

# Dual-Channel Residual Network for Hyperspectral Image Classification With Noisy Labels

Yimin Xu<sup>ID</sup>, Zhaokui Li<sup>ID</sup>, Wei Li<sup>ID</sup>, Senior Member, IEEE, Qian Du<sup>ID</sup>, Fellow, IEEE,  
Cuiwei Liu, Zhuoqun Fang<sup>ID</sup>, and Lin Zhai<sup>ID</sup>

**Abstract**—Hyperspectral image (HSI) classification has drawn increasing attention recently. However, it suffers from noisy labels that may occur during field surveys due to a lack of prior information or human mistakes. To address this issue, this article proposes a novel dual-channel residual network (DCRN) to resolve HSI classification with noisy labels. Currently, the influence of noisy labels is reduced by simply detecting and removing those anomalous samples. Different from such a specifically designed noise cleansing method, DCRN is easy to implement but highly effective. It enhances its model robustness to noisy labels to a great extent by employing a novel dual-channel structure and a noise-robust loss function. In this way, DCRN can mitigate influence from noisy labels while fully utilizing useful information from mislabeled samples for augmented training. Experiments are conducted on several hyperspectral data sets with manually generated noisy labels to demonstrate its excellent performance. The code is available at <https://github.com/Li-ZK/DCRN-2021>.

**Index Terms**—Hyperspectral image (HSI), image classification, noise-robust deep learning, noisy labels.

## I. INTRODUCTION

HYPERSPECTRAL images (HSIs) consist of rich spatial and spectral information, making their classification a very important task in remote sensing applications, since accurate classification can stimulate understanding of a complex remote scene. Thus, HSI classification is widely used in various applications, such as mineral detection [1], [2], environmental monitoring [3], [4], and agricultural analysis [5], [6].

Manuscript received November 15, 2020; revised December 29, 2020 and February 1, 2021; accepted February 3, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61922013, in part by the Liaoning Provincial Natural Science Foundation under Grant 2019-MS-254, and in part by the Beijing Natural Science Foundation under Grant JQ20021. (*Corresponding author:* Zhaokui Li.)

Yimin Xu, Zhaokui Li, and Cuiwei Liu are with the School of Computer Science, Shenyang Aerospace University, Shenyang 110136, China (e-mail: yimin.xu@duke.edu; lzk@sau.edu.cn; liucuiwei@sau.edu.cn).

Wei Li is with the School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China (e-mail: liwei089@ieee.org).

Qian Du is with the Department of Electrical and Computer Engineering, Mississippi State University, Starkville, MS 39762 USA (e-mail: du@ece.msstate.edu).

Zhuoqun Fang is with the College of Artificial Intelligence, Shenyang Aerospace University, Shenyang 110136, China (e-mail: fangzhuoqun@sau.edu.cn).

Lin Zhai is with the School of Science, Shenyang Aerospace University, Shenyang 110136, China (e-mail: zl4618816@163.com).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TGRS.2021.3057689>.

Digital Object Identifier 10.1109/TGRS.2021.3057689

Although plentiful machine learning methods [7]–[9] have achieved great success in HSI classification tasks, their performance may be degraded in the presence of noisy labels in training data. Noisy labels refer to such labels that are assigned to the wrong classes in supervised learning. Generally speaking, noisy labels are inevitable since even those most experienced annotators can make mistakes when facing some confusing samples. In HSIs, accurate annotations become increasingly difficult because of the imprecise positioning system and bias in the ground investigation. As errors are unavoidable in HSIs annotation, it becomes an essential and challenging topic to train a reliable model when the training set may contain unreliable labels.

Noisy label classification problem was first raised in computer vision applications, such as image classification [10], [11] and visual recognition [12], [13] aided by machine learning methods. To alleviate or even eliminate the influence of noisy labels, researchers in computer vision use two methods: 1) detecting and cleansing noisy labels before training and 2) directly training noise-robust models on corrupted data sets. For instance, Goldberger and Ben-Reuven [14] intended to correct noisy labels by introducing an extra layer into their deep network to estimate the transition probabilities from a true label to others. Han *et al.* [15] proposed a self-learning neural network with multiprototypes to correct noisy labels iteratively in each training epoch. Kim *et al.* [16] focused on a negative learning method to train a convolutional neural network (CNN) that “input image does not belong to this complementary label,” which does not provide wrong information as frequently as positive learning. Ma *et al.* [17] proposed a novel robust loss function to improve the robustness of deep neural networks against noisy labels. Although the aforementioned methods may achieve state-of-the-art in addressing noisy labels in related fields, these methods cannot be applied directly into HSI classification tasks in face of training samples’ deficiency and high dimensionality since the networks used in computer vision tasks are not suitable to deal with HSIs.

HSI classification with noisy labels is also a hot topic since most annotations are made by human annotators, adding to the probability of a mislabeling problem. Tu *et al.* proposed a density peak noisy label detection (DPNLD)-based method [18] to classify anomalous samples efficiently. This method is efficient in finding and correcting noisy labels to some extent, but it ignored spatial information,

which is also a vital component for HSI classification tasks. Jiang *et al.* [19] developed a random label propagation algorithm (RLPA) with superpixel constraint to reduce the level of label noise and hence stimulate more accurate supervision in the training process. Tu *et al.* [20]–[22] proposed several different methods, such as spatial density clustering (SDP) and its extension ( $K$ -SDP) to detect outliers via a spectral-spatial metric and received a better result in classification accuracy.

Although the aforementioned papers introduce new noise cleansing methods and achieve impressive results in classification accuracy, they more or less suffer from a few deficiencies. First, although those noise cleansing methods can reduce the overall noise level, they may introduce additional errors for some clean samples because of high interclass similarity compared with low intraclass similarity. Second, although noisy labels mislead training to some extent, they still have sufficient spatial and spectral information, which can be beneficial if used properly.

To address these issues, a dual-channel residual network (DCRN) is proposed to utilize a mislabeled training set directly, which can avoid noise cleansing error and make full use of the original training set. To alleviate the influence brought by noisy labels, two strategies are adopted to strengthen the robustness of the proposed framework. First, a novel deep learning network structure is developed to enlarge interclass differences, making it more separable among classes. More specifically, a dual-channel network structure with residual blocks is carefully designed to better extract separable features from both spatial channel and spectral bands. Residual blocks can reduce the influence of ambiguous features, and the fusion of spatial and spectral information further enlarges the interclass difference and ensures the robustness of our deep learning model. Moreover, the proposed DCRN is straightforward and has much fewer network parameters compared with common neural networks; hence, it can prevent problems, such as overfitting to noisy labels effectively, and have applicable generalization capability. Second, instead of using a common loss function, such as cross-entropy (CE) loss and softmax loss, a combination of reverse CE (RCE) loss and normalized CE (NCE) loss, first raised in [17], is first applied to HSI classification with label noise. The new loss function can not only reduce the influence of noisy labels by preventing overfitting in the training process but also stimulate training by a combination of active learning and passive learning strategies. To demonstrate the effectiveness of DCRN, sufficient experiments are conducted in comparison with current state-of-the-art methods, including  $K$ -SDP, DPNLD, and RLPA. Furthermore, to demonstrate the significance of our model structure, DCRN is also compared with several popular deep learning models, including the spectral-spatial residual network (SSRN) [23] and 3-D CNN [24]. To summarize, our main contribution to this article can be included in three components.

- 1) To the best of our knowledge, the deep neural network is introduced into the HSI classification problem to address noisy labels for the first time, which can offer better performance than state of the art.

- 2) A novel end-to-end network is proposed to enhance robustness by delicate model structure designation. A robust dual-channel structure, which fuses intermediate spatial and spectral information with the help of the residual block, can reduce confusing features and thus enhance the robustness of the proposed model. The proposed DCRN reaches better results not only in the overall performance but also performs better in those ambiguous classes that are usually misclassified by other methods.
- 3) To take full advantage of the informative part of mislabeled samples, a noise-robust loss function is first used to achieve better convergence with weak supervision of noisy labels. Furthermore, this novel loss function can further enhance the noise endurance of DCRN even in a high noise ratio.

The remainder of this article is structured as follows. Section II describes some related work. Section III describes the proposed method. Section IV presents related experimental results and highlights our results compared with other current results. Section V draws conclusions of this work.

## II. RELATED WORK

The proposed DCRN gains its robustness mainly from two components: a robust deep learning model and a novel robust learning metric. In this section, previous works on deep learning-based HSI classification and robust learning metrics are introduced to augment the illustration of this whole paper and further clarify our ideas and motivation.

### A. Hyperspectral Image Classification via Deep Neural Network

With the development of deep learning techniques, plentiful efficacious deep learning models have been proposed to address the HSI classification task. Deep learning-based models can be categorized into two types, i.e., spectral deep learning model and spatial-spectral deep learning model. As for spectral deep learning models, Hu *et al.* [25] proposed a 1-D deep learning framework to classify HSIs with information from their spectral bands. Mou *et al.* [26] developed a recurrent neural network (RNN) to regard the input spectral bands as a sequence and received a good result. However, these methods only use spectral features and leave out rich spatial correlations. As a result, Li *et al.* [24] proposed a 3-D CNN to utilize spatial and spectral features at the same time. Then, several 3-D neural network structures [23], [27] are proposed to accomplish HSI classification, and all perform well in clean data set.

Compared with conventional machine learning methods, these deep learning-based methods perform better since they integrate input features hierarchically and generate more separable features automatically. These features help networks obtain robustness to noisy labels to some extent. It is obvious that different network structures will lead to different latent features and result in different endurance of noisy labels. Rolnick *et al.* [28] analyzed the model robustness to noisy labels among several networks and found out that residual

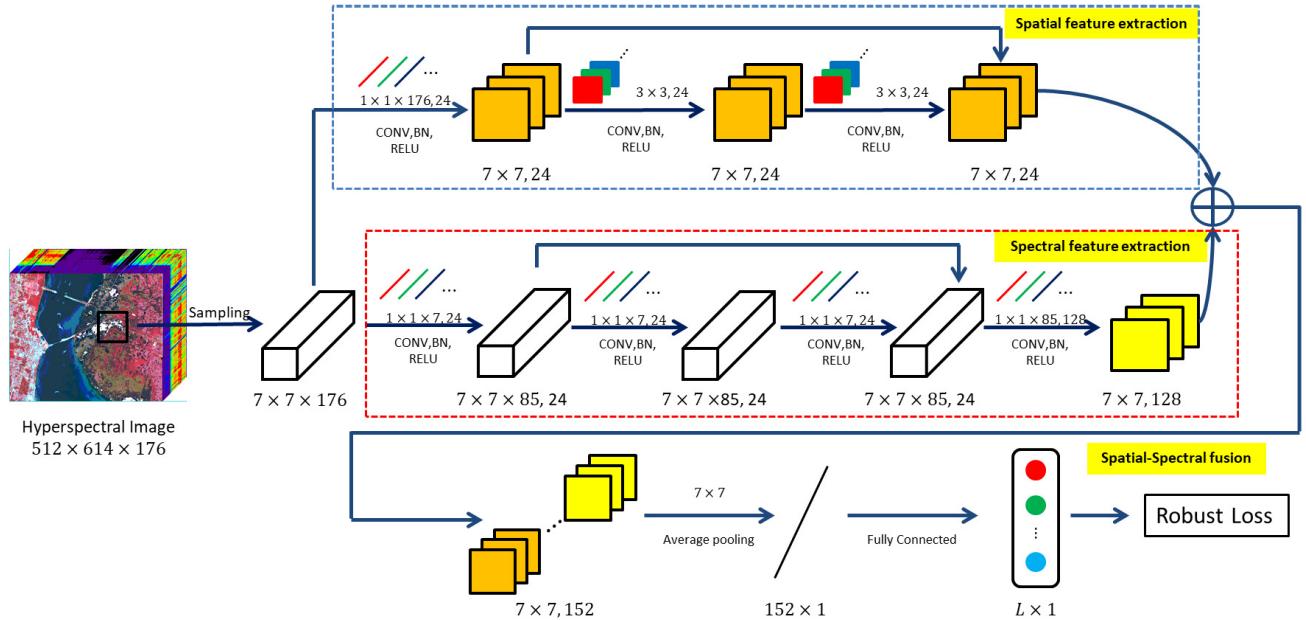


Fig. 1. DCRN that is proven to be robust to noisy labels. The network extracts spatial and spectral features, respectively, in each channel. Furthermore, both spectral channel and spatial channel contain a residual block to enhance the performance against noisy labels. After concatenating two outputs from each channel, a pooling layer and a fully connected are employed.

neural networks (ResNets) can sustain a high accuracy with increasing noise ratio. Inspired by their work, we propose a deep network structure with residual blocks to enhance the model robustness.

#### B. Training With Robust Learning Metrics

Different from label cleansing methods introduced in Section I, training directly with mislabeled samples can fully take advantage of all input samples. Currently, researchers proposed several noise-robust learning metrics/loss functions to alleviate the influence of noisy labels. Ghosh *et al.* [29] showed that loss functions, such as mean absolute error (MAE), are robust to noisy labels, while common CE is not. However, the robustness of MAE loss may cause increased difficulty in training and lead to a performance drop. Therefore, a symmetric CE loss was proposed by Wang *et al.* [30] to resolve the slow-convergence problem and mitigate the influence of noisy labels. However, the CE loss, as one of the components in the symmetric loss function, is not noise-robust. Hence, Ma *et al.* [17] proposed an APL framework that consists of two noise-robust loss functions and can boost training by active-passive learning method. Inspired by their work, NCE loss and RCE loss are combined to generate better results in both boosting training and fight against noisy labels.

### III. PROPOSED DCRN FRAMEWORK

This article aims to resolve the HSI classification task with noisy labels. Hence, a DCRN is designed to fuse spectral and spatial features and better extract discriminative features. As shown in Fig. 1, the whole framework consists of three major modules: 1) spectral feature extraction; 2) spatial feature extraction; and 3) spatial-spectral fusion. Multiple 3-D cubes

are sampled from an HSI as the input of DCRN, and  $7 \times 7 \times 176$  (7 pixels in width, 7 pixels in height, and 176 spectral bands) cubes are regarded as an example to illustrate the proposed network structure.

**Spectral Feature Extraction:** A spectral feature extraction channel (red dashed box) is employed to extract spectral features. To be specific,  $1 \times 1 \times 7$  kernels are used to learn spectral features while maintaining original spatial correlations. The spectral feature extraction module contains two convolution layers and one residual block. A batch norm (BN) layer and a ReLU activation function are applied after each convolution manipulation. In the first convolution layer, 24  $1 \times 1 \times 7$  kernels with subsampling stride (1, 1, 2) are used, which removes redundant spectral features and better focuses on those features vital in classification, mapping an original input to 24 more condensed 3-D cubes of size  $1 \times 1 \times 85$ . Then, a residual block, which is made up of two consecutive convolutional layers with 24  $1 \times 1 \times 7$  kernels, emphasizes important areas and enhances spectral robustness when dealing with noisy labels. Following the residual block, the remain cubes are convoluted with 128  $1 \times 1 \times 85$  kernels to produce 128  $7 \times 7$  2-D spatial patches. While maintaining the original spatial information, each pixel in a patch can better represent spectral information, which performs better against noisy labels, after spectral feature extraction.

**Spatial Feature Extraction:** A spatial feature extraction channel (blue dashed box) is employed to utilize spatial information around a target pixel. In this channel, 2-D convolutional layers are employed to better exploit spatial features. Similar to the spectral channel described above, a BN layer and a ReLU activation function are applied after each convolution manipulation. As shown in the blue dashed box in Fig. 1,

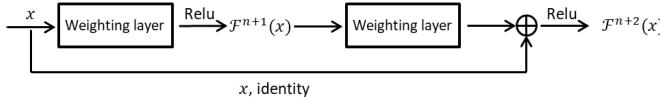


Fig. 2. Basic Structure of the residual blocks.

there are one convolutional layer and one residual block in the spatial channel. First, the input 3-D cube is convoluted with a  $1 \times 1 \times 176$  kernel to map from an HSI with multiple bands to a gray image with only one spectral band. This operation aims to force the network to focus on detecting correlations in the spatial domain. The resulting  $7 \times 7$  patches are passed to a residual block, consisting of two successive convolutional layers with  $24 \times 3 \times 3$  kernels, to preserve and emphasize important spatial information, contributing to the improvement of the overall performance in the noisy label problem.

*Spatial-Spectral Fusion:* A spatial-spectral fusion module is employed to combine information from two separate channels together. After being processed separately via both spectral and spatial channels, the output of the spectral channel is concatenated with the output of the spatial channel. Hence, after concatenating 128  $7 \times 7$  squares (spectral) and 24  $7 \times 7$  squares (spatial), 152 squares are obtained in total that is rich in both spatial and spectral information. Then, an average pooling with a  $7 \times 7$  kernel projects a 2-D matrix into a 152-D vector. Then, a fully connected layer is used to generate an output prediction,  $\hat{y} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_n]$ .

### A. Residual Blocks

ResNet was first proposed by He *et al.* [31] in 2016, and it has been proven to be one of the most successful network structures in computer vision. In this article, we find that residual blocks can improve robustness toward noisy labels by highlighting more distinctive features while reducing complexity and training time compared with Dense blocks. Hence, a residual block is used as plug-in layers to enhance the noise robustness of the traditional CNN. A common residual block is illustrated in Fig. 2.

A residual block consists of two consecutive weighting layers (such as a single convolutional layer, a convolutional layer, and a batch normalization layer)  $\mathcal{F}$  for residual learning, which can be formulated as

$$\mathcal{F}^{n+1}(x) = \sigma(\mathcal{F}(x)) \quad (1)$$

$$\mathcal{F}^{n+2}(x) = \sigma(\mathcal{F}(\mathcal{F}^{n+1}(x))) + x \quad (2)$$

where  $\sigma$  is the ReLU activation function in Fig. 2. Moreover, in the proposed DCRN, both 2-D and 3-D residual blocks with batch normalization layer are used to improve robustness since they can select more distinct spatial and spectral features, respectively.

### B. Loss Function

The core idea of the noise-robust loss function  $\mathcal{L}$  is to make sure the optimal model parameters when training with noisy

label samples are also the optimal parameters if all samples' labels are correct. Suppose a  $K$ -class data set with noisy label  $D = \{(x, y)^{(i)}\}_{i=1}^n$ , where  $x \in \mathbf{R}^{m \times n}$  denotes input feature, and  $y \in \{1, 2, \dots, K\}$  represents the label, not necessary to be correct, corresponding to the input. Meanwhile, a latent true label  $y^*$  is assigned to each sample to better demonstrate how a noise-robust loss function works. Then, the noise-robust loss function can be formulated as

$$\forall f \in \mathcal{F}, \quad \mathbf{E}_{x,y} \mathcal{L}(f^*(x), y) \leq \mathbf{E}_{x,y} \mathcal{L}(f(x), y) \quad (3)$$

where  $\mathcal{F}$  is a possible parameter space for a certain model,  $f$  is a possible solution, and  $f^*$  is global minimum of empirical risk when training with latent clean label  $y^*$ .

Specifically, noisy labels are generated by the following steps. For each class,  $m$  samples with real labels are randomly selected to form the base training set. Then,  $n$  samples from other classes are chosen evenly, and their labels are flipped to the current category. After that, all samples in each class are shuffled to make sure there are no clear clues of corrupted samples. In this way, training samples in each class can be represented as  $D^k = \{(x_1, y_1^{\text{clean}}), \dots, (x_m, y_m^{\text{clean}}), (x_{m+1}, y_{m+1}^{\text{noise}}), \dots, (x_{m+n}, y_{m+n}^{\text{noise}})\}$ , and  $k \in \{1, 2, \dots, K\}$ . Under this assumption, distribution of noise label is assumed to be conditionally independent of the input

$$P(y = k | y^* = j, x) = P(y = k | y^* = j). \quad (4)$$

Since all samples are shuffled randomly, each label has  $\eta$  possibility to be incorrect, and  $\eta$  here is equivalent to  $(n/(m+n))$ . Hence, the probability of a sample from the class  $i$  with a latent clean label  $j$  can be denoted as

$$\begin{cases} \eta_{ij} = \frac{\eta}{K-1}, & i \neq j \\ \eta_{ij} = 1 - \eta, & i = j \end{cases} \quad (5)$$

where  $\eta_{ij} = P(y = i | y^* = j)$ .

Then, a noise-robust loss function is formed as combination of NCE loss and RCE loss. NCE loss is the normalized form of CE loss, and RCE loss swaps the position of prediction ( $p(y|x)$ ) and ground truth ( $q(y|x)$ ) in normal CE loss, which can be expressed as

$$\text{NCE}(p, q) = \frac{-\sum_{k=1}^K q(k|x) \log p(k|x)}{-\sum_{j=1}^K \sum_{k=1}^K q(y=j|x) \log p(y=j|x)} \quad (6)$$

$$\text{RCE}(p, q) = -\sum_{k=1}^K p(k|x) \log q(k|x) \quad (7)$$

$$\text{Loss}(p, q) = \text{NCE}(p, q) + \text{RCE}(p, q). \quad (8)$$

These two loss functions are robust to label noise and show effectiveness in dealing with noisy labels. It is easy to prove the sum of two noise-robust loss functions is a noise-robust loss, and it becomes much easier to prove the robustness separately. First, proof of the robustness (first raised in [17]) of NCE loss will be given. Suppose that  $\mathbf{R}^n(f)$  is the empirical risk when training with corrupted labels and  $\mathbf{R}(f)$  is the empirical risk when training with true labels (i.e.,  $\mathbf{R}^n(f) = \mathbf{E}_{x,y} \mathcal{L}(f(x), y)$  and  $\mathbf{R}(f) = \mathbf{E}_{x,y^*} \mathcal{L}(f(x), y)$ ). According to the total expectation formula,

$\mathbf{R}^\eta(f) = \mathbf{E}_x \mathbf{E}_{y^*|x} \mathbf{E}_{y|x,y^*} \mathcal{L}(f(x), y)$ . By combining the chain rule and the independent assumption in (1), (6) can be transformed as

$$\begin{aligned}\mathbf{R}^\eta(f) &= \mathbf{E}_x \mathbf{E}_{y^*|x} \mathbf{E}_{y|x,y^*} \mathcal{L}(f(x), y) \\ &= \mathbf{E}_x \mathbf{E}_{y^*|x} \left[ \int P(y|x, y^*) \mathcal{L}(f(x), y) dy \right] \\ &= \mathbf{E}_x \mathbf{E}_{y^*|x} \left[ \int P(y|y^*) \mathcal{L}(f(x), y) dy \right].\end{aligned}\quad (9)$$

The probability of  $P(y|x, y^*)$  is defined in (5). Hence, (9) can be rewritten as

$$\begin{aligned}\mathbf{R}^\eta(f) &= \mathbf{E}_x \mathbf{E}_{y^*|x} \left[ (1 - \eta) \mathcal{L}(f(x), y^*) + \frac{\eta}{K-1} \sum_{k \neq y^*}^K \mathcal{L}(f(x), k) \right] \\ &= (1 - \eta) \mathcal{R}(f) + \frac{\eta}{K-1} \left( \mathbf{E}_{x,y^*} \left[ \sum_{k=1}^K (\mathcal{L}(f(x), k)) \right] - \mathcal{R}(f) \right).\end{aligned}\quad (10)$$

Since NCE is the normalized version of normal CE loss, the expectation  $\mathbf{E}_{x,y^*} [\sum_{k=1}^K (\mathcal{L}(f(x), k))] = 1$ . Thus, the simplified version of  $R^\eta(f)$  is

$$\mathbf{R}^\eta(f) = \mathcal{R}(f) \left( 1 - \frac{\eta K}{K-1} \right) + \frac{\eta}{K-1}. \quad (11)$$

Suppose that  $f^*$  is the global minimum of empirical loss of  $\mathbf{R}^\eta(f)$ , and then

$$\mathbf{R}^\eta(f^*) - \mathbf{R}^\eta(f) = (R(f^*) - R(f)) \left( 1 - \frac{\eta K}{K-1} \right) \leq 0.$$

$f^*$  is also the local minimum of  $\mathbf{R}(f)$  when the noise rate is  $\eta \leq ((K-1)/K)$ , and the NCE loss is robust to noisy labels. Replaced  $\eta$  with  $(n/(m+n))$ , the essential condition of NCE to be robust to noisy labels is  $m > (n/(K-1))$ . The proof of noise robustness of RCE loss follows merely the same procedure as the proof above. Instead of only reducing the influence of noisy labels, the combination of these two loss functions also stimulates training. While NCE aims at maximizing the probability of the right label as active learning, RCE loss, contrary to the CE loss, suppresses the probability of at least one incorrect label as passive learning. This learning procedure can further enhance the classification accuracy with noisy labels. Experiments on the influence of different loss function can be found in Section IV

#### IV. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, detailed experiments are conducted to demonstrate the effectiveness of noise-robust deep learning in HSI classification tasks. To be specific, the proposed DCRN is evaluated on the Kennedy Space Center (KSC), University of Pavia (UP), and Salinas Valley data sets. To better assess the performance of the proposed method, three common metrics, i.e., overall accuracy (OA), average accuracy (AA), and kappa efficiency (KAPPA), are used. In all three cases, all experiments are run ten times with randomly selected training data, and values of mean and standard deviation are reported.

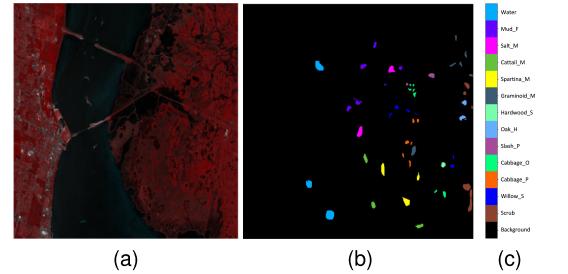


Fig. 3. Reference map for the KSC data set. (a) False-color composite image. (b) Ground-truth map. (c) Reference class name.

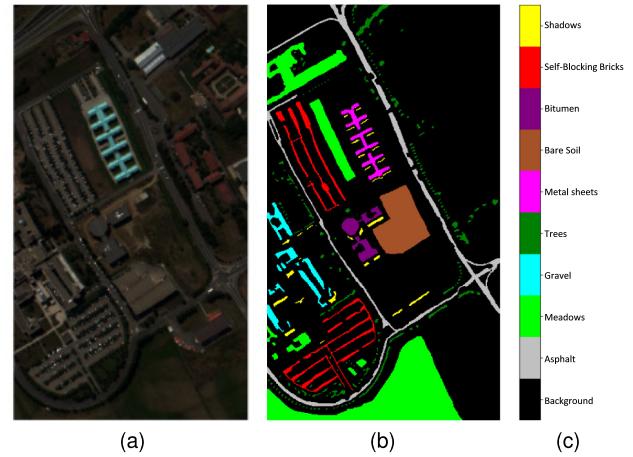


Fig. 4. Reference map for the UP data set. (a) False-color composite image. (b) Ground-truth map. (c) Reference class name.

#### A. Experimental Data Sets

To better demonstrate the performance of DCRN, three public hyperspectral data sets, i.e., KSC, UP and Salinas are used in the experiments.

- 1) *KSC*: The KSC data set was acquired by NASA AVIRIS over the KSC, Florida, USA, in 1996. The image has 512 pixels in width, 614 pixels in height, and 176 informative bands after removing water absorption and low SNR bands. For classification purposes, 13 classes representing the various land cover types that occur in this environment were defined for the site. The false-color composite of the KSC image, the ground-truth distribution map, and reference class annotation are shown in Fig. 3.
- 2) *UP*: The UP data set was acquired by the ROSIS sensor during a flight campaign over the UP in northern Italy. The image is 610 pixels in width and 240 pixels in height. The geometric resolution is 1.3 m and contains 103 spectral bands after removing noisy bands. The false-color composite of the UP image, the ground-truth distribution map, and reference class annotation are shown in Fig. 4.
- 3) *Salinas Valley*: The Salinas Valley data set was acquired by 224-band AVIRIS sensor over Salinas Valley, California, USA, and is characterized by high spatial resolution (3.7 m per pixel). The image consists of 512 lines by 217 samples. After removing 20 abundant spectral bands, 204 bands are preserved to represent 16 different classes, including vegetables, bare soils, and

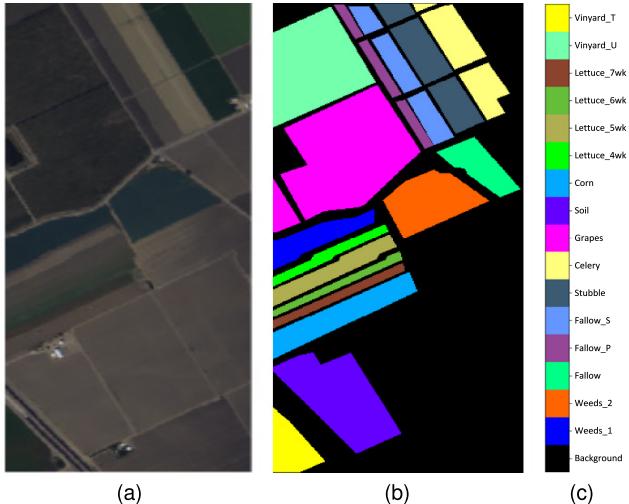


Fig. 5. Reference map for the Salinas Valley data set. (a) False-color composite image. (b) Ground-truth map. (c) Reference class name.

vineyard fields. The false-color composite of the UP image, the ground-truth distribution map, and reference class annotation are shown in Fig. 5. All reference annotations in each data set are indexed from 0(background) and counted bottom-up.

### B. Framework Settings

In these experiments, noisy labels are generated reasonably following the assumption mentioned in Section III, making sure that each class in training is made up of some clean samples and noisy samples. The DCRN is implemented in PyTorch 1.5.1, and GPU with CUDA 10.0 is used to boost the training in experiments. The batch size is chosen to be 16 because the training set is relatively small. The ADAM optimizer is adopted to harness the training process. The learning rate is set to be 0.001.

### C. Influence of Noise-Robust Loss Function

The novel noise-robust loss function is vital in training a reliable model since it prevents overfitting to noisy labels. To evaluate its influence, the KSC data set with 20 clean samples and a different number of mislabeled samples varying from 10 to 40 are used. The result is shown in Table I.

From Table I, it can be seen that the noise-robust loss function can help the training with the presence of noisy labels. The network trained with the traditional CE loss is more vulnerable to mislabeled samples. Compared with common CE loss, the proposed noise-robust loss function aids the training and improves the final classification accuracy by 2% and even 4% when labels are largely corrupted. However, a single component of the noise-robust loss, i.e., the NCE loss or the RCE loss, performs even worse than normal CE loss. Although they are robust to noisy labels, their performance is limited because of poor convergence. Hence, the combination of these two loss functions can further enhance the classification accuracy and keep robust to noisy labels.

### D. Analysis on Noise Robustness of DCRN

The proposed DCRN gains strong robustness against noisy labels from its carefully designed network structure.

TABLE I  
CLASSIFICATION ACCURACY FOR THE KSC DATA SET. CLASSIFICATION ACCURACY OBTAINED BY DCRN WITH COMMON CE LOSS, DCRN WITH NCE LOSS, DCRN WITH RCE LOSS, AND DCRN WITH NOISE ROBUST LOSS (NRL) FUNCTION WITH 20 TRUE SAMPLES AND DIFFERENT NUMBER OF MISLABELED SAMPLES

class	The number of true and noisy training samples															
	20(clean)+10(noisy)				20(clean)+20(noisy)				20(clean)+30(noisy)				20(clean)+40(noisy)			
	CE	NCE	RCE	NRL	CE	NCE	RCE	NRL	CE	NCE	RCE	NRL	CE	NCE	RCE	NRL
1	<b>98.86</b>	97.97	92.86	98.73	96.92	96.42	93.99	<b>96.94</b>	<b>97.11</b>	94.52	94.96	97.03	93.97	91.20	81.90	<b>99.59</b>
2	89.24	84.50	87.13	<b>91.12</b>	<b>93.09</b>	82.13	82.26	89.50	<b>86.80</b>	88.00	77.73	82.87	78.92	77.82	80.61	<b>85.71</b>
3	<b>99.07</b>	97.98	91.39	91.52	<b>97.46</b>	85.19	70.94	90.66	<b>96.27</b>	57.87	73.60	92.95	<b>98.90</b>	80.91	78.28	91.17
4	72.06	61.00	55.23	<b>88.70</b>	60.57	52.91	39.69	<b>76.57</b>	50.50	51.70	50.99	<b>68.96</b>	59.31	53.02	51.26	<b>63.25</b>
5	82.10	88.46	88.71	<b>94.16</b>	79.15	<b>88.82</b>	88.89	84.40	79.81	<b>88.67</b>	66.89	78.87	80.27	63.34	80.14	<b>80.43</b>
6	97.82	92.57	95.25	<b>98.09</b>	<b>92.91</b>	93.94	77.72	89.56	<b>97.31</b>	86.21	61.68	94.07	<b>88.80</b>	87.68	84.24	87.07
7	100.00	92.99	93.87	<b>100.00</b>	93.13	97.07	97.83	<b>99.76</b>	<b>93.88</b>	64.00	90.25	90.71	91.68	95.43	86.81	<b>99.06</b>
8	91.34	87.97	96.92	<b>98.88</b>	81.80	75.57	85.44	<b>94.20</b>	81.40	80.54	86.72	<b>97.31</b>	84.12	68.72	76.83	<b>93.81</b>
9	<b>95.52</b>	92.51	91.37	93.15	89.63	92.94	84.97	<b>96.07</b>	87.62	72.00	85.89	<b>88.58</b>	<b>90.43</b>	85.00	84.41	84.16
10	<b>98.33</b>	97.43	92.44	96.25	<b>91.97</b>	96.45	94.61	90.41	89.51	88.65	85.66	<b>93.38</b>	86.12	85.68	75.74	<b>90.99</b>
11	98.33	99.02	98.46	<b>99.75</b>	<b>99.05</b>	96.93	98.07	98.65	99.20	96.44	96.77	<b>99.90</b>	89.94	85.20	96.11	<b>99.90</b>
12	<b>98.84</b>	94.75	94.05	94.89	88.56	86.83	80.49	<b>92.25</b>	84.58	88.60	71.42	<b>88.18</b>	83.92	85.79	74.08	<b>85.92</b>
13	87.01	100.00	100.00	<b>100.00</b>	96.95	95.31	99.95	<b>100.00</b>	94.94	99.41	92.71	<b>100.00</b>	87.96	94.46	70.34	<b>100.00</b>
OA	94.47	93.83	92.88	<b>96.63</b>	93.11	89.85	87.80	<b>94.15</b>	89.58	86.47	84.38	<b>92.93</b>	87.79	84.52	77.97	<b>91.84</b>
(%)	(2.71)	(1.51)	(0.79)	<b>(0.98)</b>	(2.01)	(2.92)	(0.93)	<b>(1.56)</b>	(1.15)	(2.37)	(6.08)	<b>(2.00)</b>	(3.98)	(4.12)	(7.20)	<b>(2.07)</b>
AA	92.97	91.32	90.59	<b>95.79</b>	89.32	87.73	84.22	<b>92.23</b>	87.61	81.28	79.64	<b>90.21</b>	86.41	81.10	78.52	<b>89.31</b>
(%)	(3.38)	(2.56)	(2.03)	<b>(1.17)</b>	(2.76)	(2.89)	(1.83)	<b>(2.28)</b>	(1.27)	(3.93)	(3.68)	<b>(2.98)</b>	(2.73)	(4.12)	(6.28)	<b>(2.62)</b>
KAPPA	0.94	0.93	0.92	<b>0.96</b>	0.90	0.89	0.86	<b>0.93</b>	0.88	0.83	0.82	<b>0.92</b>	0.86	0.83	0.76	<b>0.91</b>
	(0.03)	(0.02)	(0.01)	<b>(0.01)</b>	(0.02)	(0.03)	(0.01)	<b>(0.02)</b>	(0.01)	(0.03)	(0.07)	<b>(0.02)</b>	(0.04)	(0.05)	(0.08)	<b>(0.02)</b>

TABLE II  
CLASSIFICATION ACCURACY FOR THE KSC DATA SET. CLASSIFICATION ACCURACY OBTAINED BY 3-D CNN, SSRN, AND DCRN WITH COMMON CE LOSS, RESPECTIVELY, WITH 20 TRUE SAMPLES AND A DIFFERENT NUMBER OF MISLABLED SAMPLES

class	The number of true and noisy training samples															
	20(clean)+10(noisy)				20(clean)+20(noisy)				20(clean)+30(noisy)				20(clean)+40(noisy)			
	CNN	SSRN	DCRN	CNN	SSRN	DCRN	CNN	SSRN	DCRN	CNN	SSRN	DCRN	CNN	SSRN	DCRN	
1	78.37	90.00	<b>98.86</b>	70.97	79.45	<b>96.92</b>	66.81	80.71	<b>97.11</b>	55.49	67.65	<b>93.97</b>	87.79	84.52	77.97	<b>91.84</b>
2	64.34	<b>89.71</b>	89.24	69.02	80.08	<b>93.99</b>	54.81	69.87	<b>86.80</b>	43.57	62.26	<b>78.92</b>	87.79	84.52	77.97	<b>88.90</b>
3	86.44	76.27	<b>99.07</b>	67.94	85.03	<b>97.46</b>	65.05	72.58	<b>96.27</b>	42.39	58.79	<b>98.90</b>	87.79	84.52	77.97	<b>91.84</b>
4	20.96	63.26	<b>72.06</b>	26.61	<b>71.07</b>	60.57	20.31	42.12	<b>50.50</b>	17.13	57.00	<b>59.31</b>	87.79	84.52	77.97	<b>88.90</b>
5	56.77	<b>89.43</b>	82.10	48.92	76.95	<b>79.15</b>	38.93	81.94	<b>79.81</b>	32.67	62.39	<b>80.27</b>	87.79	84.52	77.97	<b>88.90</b>
6	66.22	80.86	<b>97.70</b>	52.25	81.92	<b>92.91</b>	42.79	69.88	<b>97.31</b>	30.34	63.43	<b>88.80</b>	87.79	84.52	77.97	<b>88.80</b>
7	94.78	95.83	<b>100.00</b>	64.78	<b>99.05</b>	93.13	52.67	73.16	<b>93.88</b>	67.07	87.80	<b>91.68</b>	87.79	84.52	77.97	<b>88.80</b>
8	75.46	<b>93.03</b>	91.34	51.11	81.64	<b>81.80</b>	39.15	<b>82.76</b>	81.40	36.40	65.73	<b>84.12</b>	87.79	84.52	77.97	<b>88.80</b>
9	82.94	88.73	<b>95.52</b>	78.67	86.57	<b>89.63</b>	65.85	83.14	<b>87.62</b>	68.02	80.74	<b>90.43</b>	87.79	84.52	77.97	<b>88.80</b>
10	70.92	83.25	<b>98.33</b>	61.27	66.13	<b>91.97</b>	53.35	72.12	<b>89.51</b>	38.65	59.05	<b>86.12</b>	87.79	84.52	77.97	<b>88.80</b>
11	82.45	93.48	<b>98.33</b>	82.66	87.79	<b>99.05</b>	75.04	86.34	<b>99.20</b>	69.98	81.46	<b>98.94</b>	87.79	84.52	77.97	<b>88.80</b>
12	65.62	80.59	<b>98.84</b>	53.69	81.08	<b>88.56</b>	49.71	73.27	<b>84.58</b>	39.01	62.60	<b>83.92</b>	87.79	84.52	77.97	<b>88.80</b>
13	98.29	<b>99.62</b>	87.01	89.15	<b>99.37</b>	96.95	83.94	<b>94.72</b>	94.94	82.63	<b>87.78</b>	87.96	87.79	84.52	77.97	<b>88.80</b>
OA	77.10	88.61	<b>94.47</b>	68.39	84.19	<b>91.31</b>	61.34	80.00	<b>89.58</b>	54.62	71.59	<b>87.79</b>	87.79	84.52	77.97	<b>88.80</b>
(%)	(4.16)	(3.04)	<b>(2.71)</b>	(4.52)	(2.62)	<b>(2.01)</b>	(3.02)	(2.36)	<b>(1.15)</b>	(3.43)	(10.21)	<b>(3.98)</b>	87.79	84.52	77.97	<b>88.80</b>
AA	72.58	86.47	<b>92.97</b>	62.85	82.78	<b>89.32</b>	54.49	75.59	<b>87.61</b>	47.95	68.92	<b>86.41</b>	87.79	84.52	77.97	<b>88.80</b>
(%)	(3.76)	(4.23)	<b>(3.38)</b>	(4.17)	(1.83)	<b>(2.76)</b>	(4.05)	(3.48)	<b>(1.27)</b>	(3.67)	(8.56)	<b>(2.73)</b>	87.79	84.52	77.97	<b>88.80</b>
KAPPA	0.74	0.87	<b>0.94</b>	0.64	0.83	<b>0.90</b>	0.57	0.78	<b>0.88</b>	0.49	0.68	<b>0.86</b>	87.79	84.52	77.97	<b>88.80</b>
	(0.05)	(0.04)	<b>(0.03)</b>	(0.05)	(0.03)	<b>(0.02)</b>	(0.03)	(0.03)	<b>(0.01)</b>	(0.04)	(0.11)	<b>(0.04)</b>	87.79	84.52	77.97	<b>88.80</b>

By adopting a dual-channel structure, spectral information and spatial information are processed simultaneously and contribute to classification results in different extents. Furthermore, the residual blocks further emphasize more distinctive features while suppressing those ambiguous ones. The comparison between three different networks (i.e., 3-D CNN, SSRN, and DCRN), without using the noise-robust loss, is shown in Table II.

From Table II, it is obvious that the DCRN achieves far better results even without the noise-robust loss function. Although these three methods all take advantage of spatial information, their robustness against noisy labels varies greatly because of their network structures. Compared with the 3-D CNN, SSRN and DCRN both adopt residual blocks and

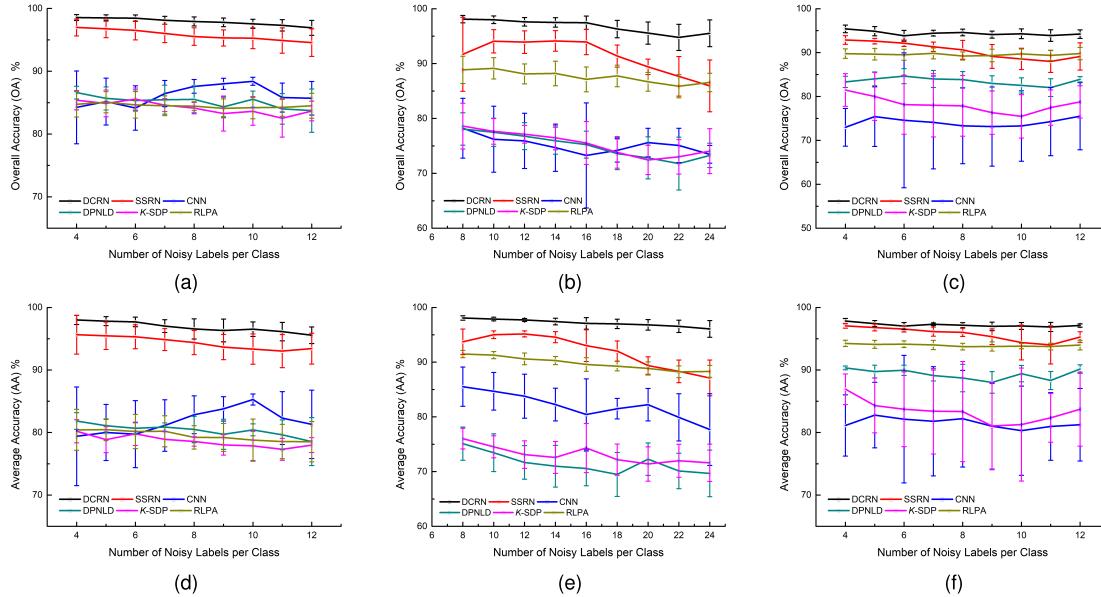


Fig. 6. Performance comparison of several methods, including DPNLD, *K*-SDP, RLPA, 3-D CNN with robust loss, SSRN with robust loss, and DCRN, in terms of OA (first row) and AA (second row). (a) and (d) are trained with the KSC data set with 24 clean samples in each class and various mislabeled samples from 4 to 12. (b) and (e) are trained with the UP data set with 52 clean samples in each class and various mislabeled samples from 8 to 24. (c) and (f) are trained with the Salinas Valley data set with 24 clean samples in each class and various mislabeled samples from 4 to 12.

TABLE III  
CLASSIFICATION ACCURACY FOR THE KSC DATA SET. CLASSIFICATION ACCURACY OBTAINED BY 3-D CNN, SSRN, AND DCRN WITH COMMON CE LOSS, RESPECTIVELY, WITH 20 TRUE SAMPLES WITHOUT NOISE SAMPLES

	CNN	SSRN	DCRN
OA	81.91	97.80	99.08
(%)	(4.71)	(0.58)	(0.48)
AA	77.73	97.28	98.33
(%)	(4.81)	(1.13)	(1.45)
KAPPA	0.80	0.98	0.99
(0.05)	(0.01)	(0.01)	

achieved better performance. Furthermore, the DCRN can achieve even higher accuracy than the SSRN method, and the dual-channel structure enables DCRN to process spatial and spectral independently in two channels, which preserves the most distinguishable features. Hence, the novel DCRN has the highest robustness and can retain high classification accuracy even trained with corrupted data sets. From Table III, the proposed DCRN also reaches the highest classification accuracy when no noisy samples are available.

#### E. Performance Evaluation

The proposed DCRN is evaluated on the KSC, University of Pavia, and Salinas data sets with the above network and parameter settings. As for KSC and Salinas data sets, each class in the training set contains 24 clean samples and several mislabeled samples from 4 to 12. For the University of Pavia data set, 52 clean samples and 8–24 corrupted samples are chosen in each class to form a valid training set. Five methods (DPNLD [18], *K*-SDP [20], RLPA [19], 3-D CNN [24] with robust loss, and SSRN [23] with robust loss) are compared

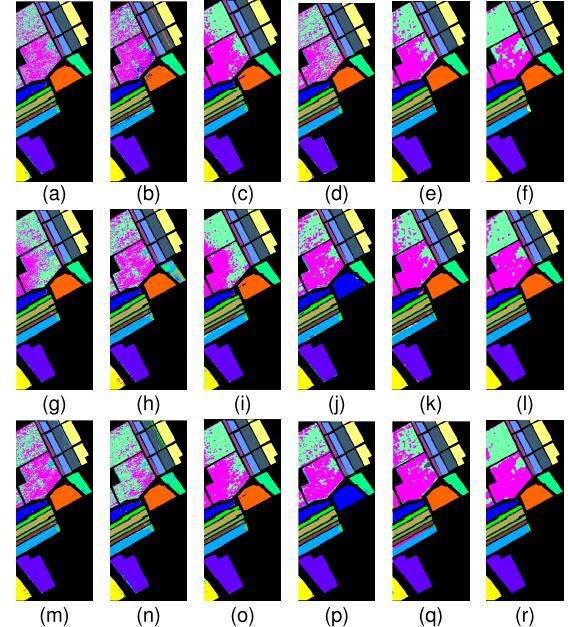


Fig. 7. Classification map for the Salinas Valley data set. Classification map obtained by DPNLD (first column), *K*-SDP (second column), RLPA with ELM classifier (third column), 3-D CNN with robust loss function (forth column), SSRN with robust loss function (fifth column), and DCRN (last column). (a)–(f) Four mislabeled samples for each class. (g)–(l) Eight mislabeled samples for each class. (m)–(r) 12 mislabeled samples for each class.

with the DCRN. Among these five methods, DPNLD, *K*-SDP, and RLPA are considered states of the art, which aims at detecting and removing mislabeled samples. 3-D CNN and SSRN are common networks in HSI classification without the consideration of noisy labels and are trained with the noise-robust loss function to show the robustness of DCRN.

TABLE IV

CLASSIFICATION ACCURACY FOR THE UP DATA SET. CLASSIFICATION ACCURACY OBTAINED BY THE DPNLD,  $K$ -SDP, RLPA, 3-D CNN WITH ROBUST LOSS, SSRN WITH ROBUST LOSS, AND DCRN WITH 52 TRUE SAMPLES AND A DIFFERENT NUMBER OF MISLABLED SAMPLES

class	The number of true and noisy training samples																	
	52(clean)+8(noisy)						52(clean)+16(noisy)						52(clean)+24(noisy)					
	DPNLD	$K$ -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	$K$ -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	$K$ -SDP	RLPA	CNN	SSRN	DCRN
1	90.17	92.54	81.63	76.50	96.00	<b>96.52</b>	89.79	90.34	81.20	73.51	88.87	<b>94.75</b>	86.75	86.43	<b>80.86</b>	73.33	83.77	89.43
2	93.48	94.62	89.73	70.69	94.61	<b>96.86</b>	92.39	92.38	87.14	68.06	95.41	<b>96.05</b>	92.63	93.37	<b>85.93</b>	72.88	92.95	93.65
3	62.15	65.28	88.76	68.45	92.17	<b>95.72</b>	55.58	58.21	87.59	73.68	83.65	<b>95.05</b>	54.87	54.59	85.23	71.16	76.73	<b>94.18</b>
4	76.95	66.74	91.13	91.95	<b>97.99</b>	97.42	69.55	69.73	85.75	92.82	94.35	<b>96.05</b>	69.29	62.61	83.09	91.62	94.39	<b>96.67</b>
5	95.60	87.34	93.96	98.11	<b>99.98</b>	99.69	84.14	88.79	93.01	97.62	<b>99.99</b>	99.92	76.17	87.87	84.62	98.25	<b>99.68</b>	99.25
6	54.53	61.39	91.72	74.78	96.55	<b>99.47</b>	53.90	61.14	88.72	78.07	95.96	<b>96.98</b>	48.81	52.11	86.60	70.82	91.01	<b>97.28</b>
7	45.09	59.20	98.00	89.91	98.26	<b>98.99</b>	44.78	49.30	97.54	90.39	95.33	<b>98.95</b>	44.64	44.08	97.38	88.35	88.32	<b>98.69</b>
8	76.27	78.33	90.93	81.07	87.85	<b>95.68</b>	74.31	76.52	90.20	80.96	82.65	<b>95.70</b>	73.41	75.10	89.85	73.49	73.59	<b>92.43</b>
9	89.58	89.71	98.44	97.20	99.71	<b>99.71</b>	84.15	82.28	97.43	98.20	99.45	<b>99.65</b>	81.59	84.17	97.53	96.72	99.39	<b>99.44</b>
OA (%)	77.47 (2.20)	80.69 (2.71)	89.42 (1.87)	76.34 (4.11)	94.98 (2.25)	<b>97.21</b> (0.97)	75.31 (3.43)	78.59 (3.93)	87.29 (2.40)	75.41 (6.89)	91.94 (2.24)	<b>96.16</b> (1.18)	71.70 (3.30)	76.66 (3.11)	85.86 (2.47)	75.73 (3.73)	89.16 (3.89)	<b>94.01</b> (2.22)
AA (%)	75.98 (1.07)	77.24 (1.70)	91.59 (0.46)	83.18 (3.45)	95.90 (1.13)	<b>97.79</b> (0.43)	72.07 (3.84)	74.30 (2.22)	89.84 (1.27)	83.70 (3.75)	92.85 (2.34)	<b>97.01</b> (0.65)	69.80 (4.23)	71.15 (3.04)	87.90 (0.97)	81.85 (3.04)	88.87 (4.07)	<b>95.74</b> (3.10)
KAPPA	0.71	0.75	0.86	0.70	0.93	<b>0.96</b>	0.69	0.73	0.84	0.69	0.91	<b>0.95</b>	0.65	0.71	0.82	0.69	0.86	<b>0.92</b>
	0.25	(0.03)	(0.02)	(0.05)	(0.03)	<b>(0.01)</b>	(0.04)	(0.04)	(0.03)	(0.08)	(0.03)	<b>(0.02)</b>	(0.04)	(0.03)	(0.04)	(0.05)	<b>(0.03)</b>	

TABLE V

CLASSIFICATION ACCURACY FOR THE KSC DATA SET. CLASSIFICATION ACCURACY OBTAINED BY THE DPNLD,  $K$ -SDP, RLPA, 3-D CNN WITH ROBUST LOSS, SSRN WITH ROBUST LOSS, AND DCRN WITH 24 TRUE SAMPLES AND A DIFFERENT NUMBER OF MISLABLED SAMPLES

class	The number of true and noisy training samples																	
	24(clean)+4(noisy)						24(clean)+8(noisy)						24(clean)+12(noisy)					
	DPNLD	$K$ -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	$K$ -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	$K$ -SDP	RLPA	CNN	SSRN	DCRN
1	95.67	96.13	92.30	90.63	98.44	<b>99.12</b>	94.90	95.78	96.04	91.46	95.14	<b>97.79</b>	95.97	94.74	96.87	91.11	96.22	<b>98.63</b>
2	85.14	84.91	83.32	86.95	96.28	<b>98.68</b>	85.28	80.62	82.48	85.57	95.80	<b>97.02</b>	82.14	76.71	80.27	81.84	92.85	<b>96.26</b>
3	85.16	88.35	72.46	86.72	93.42	<b>98.99</b>	86.25	80.91	76.49	<b>92.57</b>	84.70	90.83	83.09	82.34	74.26	88.80	90.34	<b>97.35</b>
4	63.20	66.03	59.99	42.39	80.41	<b>89.35</b>	64.28	65.30	52.47	45.89	82.98	<b>91.49</b>	60.41	61.81	59.37	44.71	82.79	<b>88.67</b>
5	63.05	66.09	44.96	80.25	94.72	<b>95.10</b>	63.11	68.87	49.47	79.54	90.59	<b>95.61</b>	60.67	65.25	37.88	72.45	93.02	<b>92.18</b>
6	57.16	62.13	52.69	62.76	96.14	<b>99.35</b>	58.75	57.86	47.71	73.02	87.38	<b>99.14</b>	57.74	63.30	50.86	70.58	93.25	<b>99.74</b>
7	72.01	70.71	98.71	99.74	99.48	<b>100.00</b>	67.12	67.85	99.04	99.60	98.08	<b>97.13</b>	66.68	60.63	95.63	77.91	99.42	<b>99.73</b>
8	79.17	81.80	92.49	72.58	99.15	<b>99.95</b>	77.32	82.79	91.97	75.78	97.20	<b>98.93</b>	69.98	79.33	92.31	64.54	96.92	<b>97.78</b>
9	88.37	90.76	87.83	81.43	95.97	<b>99.92</b>	84.76	93.60	87.05	78.48	98.06	<b>99.57</b>	83.57	90.61	86.67	82.26	96.91	<b>96.79</b>
10	94.21	82.49	87.88	90.31	97.00	<b>99.71</b>	87.92	90.22	87.85	91.79	95.05	<b>97.68</b>	90.78	90.35	85.50	86.37	92.34	<b>94.81</b>
11	93.22	93.98	96.19	95.92	99.69	<b>99.77</b>	86.54	83.32	95.20	97.88	<b>99.25</b>	98.65	91.00	85.24	95.24	95.68	99.50	<b>98.72</b>
12	88.79	91.46	65.12	77.17	95.66	<b>97.85</b>	89.17	92.30	63.95	81.84	93.69	<b>96.02</b>	87.51	90.97	64.87	65.74	92.46	<b>95.63</b>
13	98.43	96.98	95.59	88.28	100.00	<b>100.00</b>	98.81	97.46	95.82	90.00	100.00	<b>100.00</b>	98.29	99.39	95.42	96.46	99.49	<b>100.00</b>
OA (%)	86.91 (1.94)	87.04 (1.60)	84.12 (2.24)	82.98 (6.56)	96.91 (1.02)	<b>98.84</b> (0.97)	85.38 (1.78)	86.34 (1.15)	84.20 (1.17)	84.96 (6.41)	95.28 (2.28)	<b>97.70</b> (1.18)	84.36 (1.67)	85.62 (3.45)	84.08 (2.01)	82.36 (5.32)	95.38 (1.56)	<b>97.36</b> (2.22)
AA (%)	81.81 (2.36)	82.45 (1.72)	79.20 (3.17)	81.16 (4.89)	95.88 (1.53)	<b>98.29</b> (0.43)	80.32 (1.41)	81.30 (1.47)	78.89 (2.72)	83.42 (3.43)	93.69 (3.44)	<b>96.91</b> (0.65)	79.06 (2.27)	80.05 (2.58)	78.09 (2.87)	78.34 (6.38)	94.27 (2.03)	<b>96.71</b> (1.30)
KAPPA	0.85	0.855	0.82	0.81	0.97	<b>0.99</b>	0.84	0.848	0.83	0.83	0.95	<b>0.97</b>	0.83	0.84	0.82	0.80	0.95	<b>0.97</b>
	(0.02)	(0.00)	(0.02)	(0.07)	(0.01)	<b>(0.01)</b>	(0.02)	(0.01)	(0.01)	(0.07)	(0.03)	<b>(0.02)</b>	(0.02)	(0.04)	(0.02)	(0.06)	(0.02)	<b>(0.03)</b>

The classification results are shown in Fig. 6, which presents the mean and standard deviation of OA and AA when training with a different number of noisy label samples. Although using the robust loss function, traditional CNN performs poorly in all three data sets and has a higher variance, meaning that it may suffer from instability. However, deep neural network models with residual blocks (SSRN and DCRN) always have better performance and are more stable compared with those noise cleansing methods. Furthermore, the DCRN generates higher OA and AA in comparison with the SSRN and other methods. This result demonstrates that a deep learning framework can offer much better performance with the combination of a robust network model and a robust loss function. In addition, the result shows that the DCRN has higher robustness compared with the common CNN and the simple residual network (SSRN).

Classification results in the UP data set are presented in Table IV. The DCRN outperforms those noise cleansing methods by over 7.5% and outperforms SSRN by 2% on average. More importantly, it can achieve high classification accuracy in some confusing classes, such as class 3 (Gravel), class 6 (Bare Soil), and class 8 (Self-Blocking Bricks). Similar results can also be drawn from other testing data sets from Tables V and VI. In the University of Pavia data set, the accuracy is improved by nearly 20% compared with  $K$ -SDP. In the Salinas data set, the OA is improved by 10% under all noise levels.

To better visualize the classification performance, a classification map in the Salinas data set is presented in Fig. 7, and only areas with the ground truth are shown in this map to better compare different methods. It is obvious that the classification map from the last column has a smoother area in

TABLE VI

CLASSIFICATION ACCURACY FOR THE SALINAS DATA SET. CLASSIFICATION ACCURACY OBTAINED BY THE DPNLD, *K*-SDP, RLPA, 3-D CNN WITH ROBUST LOSS, SSRN WITH ROBUST LOSS, AND DCRN WITH 24 TRUE SAMPLES AND A DIFFERENT NUMBER OF MISLABLED SAMPLES

class	The number of true and noisy training samples																	
	24(clean)+4(noisy)						24(clean)+8(noisy)						24(clean)+12(noisy)					
	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN
1	99.58	99.03	97.53	20.11	98.05	<b>99.73</b>	99.37	95.56	97.42	79.44	<b>99.67</b>	99.54	98.88	97.77	97.31	39.97	98.74	<b>99.69</b>
2	98.95	98.89	98.93	69.72	100.00	<b>100.00</b>	99.02	98.47	98.24	9.99	99.93	<b>99.97</b>	98.72	98.62	98.36	29.93	99.64	<b>100.00</b>
3	90.25	87.07	99.30	85.92	99.83	<b>99.99</b>	86.78	88.54	98.61	94.00	99.81	<b>99.84</b>	89.27	90.71	97.88	87.80	99.54	<b>99.84</b>
4	96.61	97.09	98.83	99.00	99.39	<b>99.61</b>	93.93	96.92	98.80	99.27	99.36	<b>99.51</b>	97.22	97.43	98.78	89.06	98.74	<b>99.64</b>
5	<b>98.60</b>	98.15	95.09	95.94	98.52	96.96	98.47	95.22	93.71	94.87	<b>98.49</b>	94.37	<b>98.61</b>	97.05	94.16	96.20	96.12	96.30
6	100.00	<b>100.00</b>	99.25	99.55	99.66	99.94	99.42	<b>100.00</b>	96.98	98.63	99.81	99.78	99.26	<b>100.00</b>	96.58	98.94	99.66	99.62
7	98.17	97.44	98.65	58.27	99.70	<b>99.93</b>	98.68	96.03	98.67	78.11	98.79	<b>99.82</b>	97.14	97.18	98.65	68.63	99.19	<b>99.75</b>
8	73.20	72.60	76.32	61.67	<b>84.21</b>	81.18	72.12	70.88	77.89	37.06	79.00	<b>81.17</b>	71.16	69.98	77.01	46.88	65.21	<b>77.12</b>
9	99.39	98.63	99.27	87.78	99.75	<b>99.89</b>	99.20	99.05	99.18	95.65	99.48	<b>99.89</b>	99.29	98.77	99.23	92.14	<b>99.45</b>	99.04
10	75.65	71.30	90.20	77.55	94.51	<b>96.38</b>	79.61	82.45	89.51	81.97	91.95	<b>96.78</b>	78.93	77.50	89.31	87.32	91.32	<b>95.71</b>
11	86.51	91.28	93.02	77.94	98.41	<b>99.69</b>	90.44	83.33	92.70	94.32	98.26	<b>99.48</b>	87.40	82.35	93.21	93.99	97.68	<b>99.34</b>
12	94.60	92.88	99.45	99.46	<b>99.96</b>	99.92	91.60	91.37	99.31	98.76	<b>100.00</b>	99.98	94.18	88.99	99.40	99.48	99.76	<b>99.82</b>
13	94.44	91.35	98.58	98.57	<b>99.82</b>	99.63	87.54	89.26	98.69	98.96	<b>99.53</b>	99.29	90.76	91.10	98.66	98.10	<b>99.87</b>	99.49
14	88.59	93.47	93.86	85.98	<b>99.31</b>	99.10	89.81	91.47	93.56	94.42	<b>99.05</b>	98.99	91.43	83.74	92.54	96.18	<b>98.69</b>	98.58
15	53.54	53.63	73.54	69.87	76.11	<b>89.56</b>	56.00	53.00	72.56	75.46	83.48	<b>89.62</b>	51.96	52.92	72.26	70.13	77.32	<b>92.58</b>
16	86.44	96.46	96.59	93.06	96.84	<b>98.26</b>	91.02	96.04	95.36	94.01	<b>97.56</b>	96.80	92.51	92.35	93.61	94.57	<b>96.64</b>	96.42
OA (%)	83.86 (1.58)	84.02 (1.54)	89.78 (1.01)	75.60 (6.39)	92.74 (1.93)	<b>94.15</b> (1.49)	84.37 (1.21)	83.81 (1.11)	72.43 (1.50)	92.47 (0.88)	<b>93.95</b> (1.30)	83.35 (1.82)	83.50 (1.58)	89.22 (1.84)	72.63 (7.85)	88.47 (4.66)	<b>93.40</b> (1.88)	
AA (%)	89.66 (1.57)	89.96 (1.82)	94.27 (0.48)	80.02 (7.07)	96.50 (1.23)	<b>97.49</b> (0.52)	89.56 (0.84)	89.22 (0.92)	93.82 (0.80)	82.81 (3.76)	96.51 (0.57)	<b>97.18</b> (0.47)	89.79 (1.28)	88.53 (0.81)	93.56 (0.74)	80.58 (6.27)	94.85 (1.85)	<b>97.06</b> (0.56)
KAPPA	0.82 (0.02)	0.82 (0.02)	0.89 (0.02)	0.73 (0.07)	0.91 (0.02)	<b>0.94</b> (0.02)	0.83 (0.01)	0.820 (0.01)	0.89 (0.07)	0.70 (0.01)	0.92 (0.01)	<b>0.93</b> (0.01)	0.82 (0.02)	0.817 (0.02)	0.88 (0.08)	0.70 (0.05)	0.87 (0.05)	<b>0.93</b> (0.02)

TABLE VII

CLASSIFICATION ACCURACY FOR THREE DISTINCT DATA SETS. CLASSIFICATION ACCURACY OBTAINED BY THE DPNLD, *K*-SDP, RLPA, 3-D CNN WITH ROBUST LOSS, SSRN WITH ROBUST LOSS, AND DCRN WITH FEW TRUE SAMPLES AND A DIFFERENT NUMBER OF MISLABLED SAMPLES

Metrics	The number of true and noisy training samples(KSC)																	
	8(clean)+4(noisy)						8(clean)+8(noisy)						8(clean)+12(noisy)					
	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN
OA (%)	74.24 (3.82)	72.53 (3.97)	76.04 (5.71)	76.85 (5.65)	73.74 (8.01)	<b>91.60</b> (1.92)	71.30 (2.28)	70.03 (4.50)	72.35 (5.26)	74.91 (3.40)	69.31 (2.40)	<b>87.42</b> (3.03)	64.18 (4.08)	63.36 (10.67)	73.08 (2.87)	60.29 (6.83)	57.82 (3.12)	<b>83.44</b> (3.72)
AA (%)	70.34 (3.56)	67.55 (3.50)	72.33 (4.83)	72.96 (6.73)	69.71 (7.39)	<b>89.92</b> (1.62)	66.27 (1.35)	65.00 (3.26)	68.26 (5.39)	69.75 (4.16)	65.29 (2.95)	<b>84.99</b> (3.50)	62.29 (2.87)	58.05 (6.73)	66.80 (3.94)	54.09 (5.16)	53.39 (2.80)	<b>78.97</b> (5.18)
KAPPA	0.71 (0.04)	0.70 (0.04)	0.74 (0.06)	0.74 (0.06)	0.71 (0.09)	<b>0.91</b> (0.02)	0.68 (0.02)	0.67 (0.05)	0.70 (0.04)	0.72 (0.04)	0.66 (0.03)	<b>0.86</b> (0.03)	0.60 (0.04)	0.60 (0.11)	0.71 (0.03)	0.56 (0.03)	0.53 (0.04)	<b>0.81</b> (0.04)
Metrics	The number of true and noisy training samples(Univeristy of Pavia)																	
	20(clean)+8(noisy)						20(clean)+16(noisy)						20(clean)+24(noisy)					
	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN
OA (%)	61.96 (3.88)	61.34 (3.03)	74.82 (3.27)	69.19 (1.75)	77.90 (3.59)	<b>90.77</b> (3.32)	52.61 (7.94)	56.40 (4.13)	68.84 (4.52)	64.27 (3.65)	64.38 (4.12)	<b>84.58</b> (2.12)	52.37 (5.48)	50.98 (7.19)	60.27 (2.98)	62.08 (2.77)	47.55 (3.96)	<b>81.24</b> (3.96)
AA (%)	62.90 (3.95)	64.78 (2.59)	79.83 (1.34)	76.62 (1.68)	81.80 (3.97)	<b>94.05</b> (1.55)	58.92 (7.79)	60.22 (5.73)	73.55 (2.26)	71.73 (1.35)	65.64 (2.12)	<b>88.68</b> (2.57)	56.77 (5.95)	57.91 (2.75)	66.18 (2.84)	68.25 (3.56)	50.71 (2.62)	<b>84.23</b> (2.75)
KAPPA	0.54 (0.04)	0.53 (0.03)	0.69 (0.05)	0.61 (0.02)	0.72 (0.04)	<b>0.88</b> (0.04)	0.44 (0.03)	0.48 (0.05)	0.63 (0.04)	0.56 (0.03)	0.55 (0.03)	<b>0.80</b> (0.03)	0.44 (0.04)	0.42 (0.04)	0.54 (0.04)	0.53 (0.03)	0.36 (0.04)	<b>0.75</b> (0.05)
Metrics	The number of true and noisy training samples(Salinas Valley)																	
	8(clean)+4(noisy)						8(clean)+8(noisy)						8(clean)+12(noisy)					
	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN	DPNLD	<i>K</i> -SDP	RLPA	CNN	SSRN	DCRN
OA (%)	77.79 (1.67)	73.35 (6.46)	83.88 (1.62)	72.96 (6.80)	81.57 (3.53)	<b>88.80</b> (0.93)	77.49 (2.64)	58.36 (7.09)	80.38 (2.78)	67.33 (1.11)	73.30 (5.87)	<b>84.49</b> (3.15)	75.70 (4.25)	55.94 (4.46)	78.21 (4.77)	62.25 (10.76)	56.54 (12.14)	<b>82.86</b> (4.53)
AA (%)	84.35 (3.72)	77.85 (5.84)	89.39 (1.38)	79.62 (4.82)	86.27 (2.90)	<b>93.52</b> (1.37)	82.94 (2.95)	57.74 (9.55)	85.51 (2.38)	71.98 (4.02)	77.91 (3.60)	<b>90.85</b> (2.14)	80.67 (1.64)	54.65 (6.96)	82.85 (3.29)	67.86 (10.76)	58.75 (7.79)	<b>89.28</b> (1.66)
KAPPA	0.76 (0.02)	0.71 (0.07)	0.83 (0.02)	0.70 (0.07)	0.80 (0.04)	<b>0.88</b> (0.01)	0.75 (0.03)	0.54 (0.08)	0.79 (0.03)	0.63 (0.02)	0.71 (0.06)	<b>0.83</b> (0.03)	0.73 (0.04)	0.51 (0.04)	0.76 (0.05)	0.59 (0.11)	0.53 (0.12)	<b>0.82</b> (0.05)

almost all areas compared with other corresponding columns. Furthermore, the classification map, generated by the DCRN, shows higher accuracy in grapes and vinyard\_U.

#### F. Performance With Few Clean Samples in Each Class

The number of clean samples is vital in a noisy label classification problem, and the proposed DCRN framework can work well even when few clean samples are available.

As shown in Table VII, the proposed DCRN still has a relatively good performance when the number of clean samples is limited in each class. The SSRN model degrades greatly because it has more parameters and a weaker structure compared to the DCRN. Moreover, the DCRN shows a slow degradation in performance compared with the SSRN model with an increasing number of noisy samples. Thus, the DCRN has higher stability compared with other methods.

## V. CONCLUSION

This article proposed a new idea in dealing with label noise in the HSI classification problem. As far as we know, the deep neural network was introduced to solve this problem for the first time in this article. The proposed DCRN fully utilized mislabeled samples instead of simply removing or correcting them. Furthermore, the DCRN took advantage of the property of residual blocks and reached state-of-the-art performance with the aid of a novel loss function, which is robust to noisy labels. The DCRN was tested on three distinct data sets and outperformed other comparative methods.

## REFERENCES

- [1] N. Yokoya, J. Chan, and K. Segl, "Potential of resolution-enhanced hyperspectral data for mineral mapping using simulated EnMAP and Sentinel-2 images," *Remote Sens.*, vol. 8, no. 3, p. 172, Feb. 2016.
- [2] J. Li, J. M. Bioucas-Dias, A. Plaza, and L. Liu, "Robust collaborative nonnegative matrix factorization for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6076–6090, Oct. 2016.
- [3] X. Yang and Y. Yu, "Estimating soil salinity under various moisture conditions: An experimental study," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 5, pp. 2525–2533, May 2017.
- [4] C. Xu, W. Zeng, J. Huang, J. Wu, and W. van Leeuwen, "Prediction of soil moisture content and soil salt concentration from hyperspectral laboratory and field data," *Remote Sens.*, vol. 8, no. 1, p. 42, Jan. 2016.
- [5] L. Liang *et al.*, "Estimation of crop LAI using hyperspectral vegetation indices and a hybrid inversion method," *Remote Sens. Environ.*, vol. 165, pp. 123–134, Aug. 2015.
- [6] L. Liang *et al.*, "Estimation of leaf nitrogen content in wheat using new hyperspectral indices and a random forest regression algorithm," *Remote Sens.*, vol. 10, no. 12, p. 1940, Dec. 2018.
- [7] J. Ham, Y. Chen, M. M. Crawford, and J. Ghosh, "Investigation of the random forest framework for classification of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 3, pp. 492–501, Mar. 2005.
- [8] H. Li, Y. Song, and C. L. P. Chen, "Hyperspectral image classification based on multiscale spatial information fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 9, pp. 5302–5312, Sep. 2017.
- [9] C. Tao, H. Pan, Y. Li, and Z. Zou, "Unsupervised spectral-spatial feature learning with stacked sparse autoencoder for hyperspectral imagery classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 12, pp. 2438–2442, Dec. 2015.
- [10] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2691–2699.
- [11] K.-H. Lee, X. He, L. Zhang, and L. Yang, "CleanNet: Transfer learning for scalable image classifier training with label noise," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5447–5456.
- [12] Y. Li, J. Yang, Y. Song, L. Cao, J. Luo, and L.-J. Li, "Learning from noisy labels with distillation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1910–1918.
- [13] W. Zhang, Y. Wang, and Y. Qiao, "MetaCleaner: Learning to hallucinate clean representations for noisy-labeled visual recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7373–7382.
- [14] J. Goldberger and E. Ben-Reuven, "Training deep neural-networks using a noise adaptation layer," in *Proc. ICLR*, 2017, pp. 1–9.
- [15] J. Han, P. Luo, and X. Wang, "Deep self-learning from noisy labels," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5138–5147.
- [16] Y. Kim, J. Yim, J. Yun, and J. Kim, "NLNL: Negative learning for noisy labels," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 101–110.
- [17] X. Ma, H. Huang, Y. Wang, S. Romano, S. Erfani, and J. Bailey, "Normalized loss functions for deep learning with noisy labels," 2020, *arXiv:2006.13554*. [Online]. Available: <http://arxiv.org/abs/2006.13554>
- [18] B. Tu, X. Zhang, X. Kang, G. Zhang, and S. Li, "Density peak-based noisy label detection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 3, pp. 1573–1584, Mar. 2019.
- [19] J. Jiang, J. Ma, Z. Wang, C. Chen, and X. Liu, "Hyperspectral image classification in the presence of noisy labels," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 851–865, Feb. 2019.
- [20] B. Tu, X. Zhang, X. Kang, J. Wang, and J. A. Benediktsson, "Spatial density peak clustering for hyperspectral image classification with noisy labels," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 5085–5097, Jul. 2019.
- [21] B. Tu, C. Zhou, X. Liao, Z. Xu, Y. Peng, and X. Ou, "Hierarchical structure-based noisy labels detection for hyperspectral image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 2183–2199, 2020.
- [22] B. Tu, C. Zhou, D. He, S. Huang, and A. Plaza, "Hyperspectral classification with noisy label detection via superpixel-to-pixel weighting distance," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 6, pp. 4116–4131, Jun. 2020.
- [23] 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.
- [24] Y. Li, H. Zhang, and Q. Shen, "Spectral-spatial classification of hyperspectral imagery with 3D convolutional neural network," *Remote Sens.*, vol. 9, no. 1, p. 67, Jan. 2017.
- [25] W. Hu, Y. Huang, L. Wei, F. Zhang, and H. Li, "Deep convolutional neural networks for hyperspectral image classification," *J. Sensors*, vol. 2015, pp. 1–12, Jan. 2015.
- [26] 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.
- [27] Y. Chen, H. Jiang, C. Li, X. Jia, and P. Ghamisi, "Deep feature extraction and classification of hyperspectral images based on convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 10, pp. 6232–6251, Oct. 2016.
- [28] D. Rolnick, A. Veit, S. Belongie, and N. Shavit, "Deep learning is robust to massive label noise," 2017, *arXiv:1705.10694*. [Online]. Available: <http://arxiv.org/abs/1705.10694>
- [29] A. Ghosh, H. Kumar, and P. S. Sastry, "Robust loss functions under label noise for deep neural networks," 2017, *arXiv:1712.09482*. [Online]. Available: <http://arxiv.org/abs/1712.09482>
- [30] Y. Wang, X. Ma, Z. Chen, Y. Luo, J. Yi, and J. Bailey, "Symmetric cross entropy for robust learning with noisy labels," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 322–330.
- [31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.



**Yimin Xu** received the M.S. degree in electrical and computer engineering from Duke University, Durham, NC, USA in May 2020.

He interns at the School of Computer Science, Shenyang Aerospace University, Shenyang, China. His research interests include computer vision, deep learning, weak-supervised learning, and pattern recognition.



**Zhaokui Li** received the M.S. degree in computer application from Liaoning University, Shenyang, China, in 2003, and the Ph.D. degree in computer software and theory from Wuhan University, Wuhan, China, in 2014.

He is a Professor with the School of Computer, Shenyang Aerospace University, Shenyang. His research interests include hyperspectral image analysis, computer vision, and machine learning.



**Wei Li** (Senior Member, IEEE) received the B.E. degree in telecommunications engineering from Xidian University, Xi'an, China, in 2007, the M.S. degree in information science and technology from Sun Yat-sen University, Guangzhou, China, in 2009, and the Ph.D. degree in electrical and computer engineering from Mississippi State University, Starkville, MS, USA, in 2012.

Subsequently, he spent one year as a Post-Doctoral Researcher at the University of California at Davis, Davis, CA, USA. He is a Professor with the School

of Information and Electronics, Beijing Institute of Technology, Beijing, China. His research interests include hyperspectral image analysis, pattern recognition, and data compression.

Dr. Li is also an Associate Editor of the IEEE SIGNAL PROCESSING LETTERS (SPL) and the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING (JSTARS).



**Qian Du** (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Maryland, Baltimore, MD, USA, in 2000.

She is the Bobby Shackouls Professor with the Department of Electrical and Computer Engineering, Mississippi State University, Starkville, MS, USA. Her research interests include hyperspectral remote sensing image analysis and applications, pattern classification, data compression, and neural networks.

Dr. Du is also a fellow of the SPIE-International Society for Optics and Photonics. She received the 2010 Best Reviewer Award from the IEEE Geoscience and Remote Sensing Society. She was the Co-Chair of the Data Fusion Technical Committee of the IEEE Geoscience and Remote Sensing Society from 2009 to 2013 and the Chair of the Remote Sensing and Mapping Technical Committee of the International Association for Pattern Recognition from 2010 to 2014. She is also the General Chair of the 4th IEEE GRSS Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Shanghai, China, in 2012. She has served as an Associate Editor for the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING, the *Journal of Applied Remote Sensing*, and the IEEE SIGNAL PROCESSING LETTERS. Since 2016, she has been the Editor-in-Chief of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING.



**Cuiwei Liu** received the B.S. and Ph.D. degrees from the Beijing Institute of Technology, Beijing, China, in 2009 and 2015, respectively.

She is an Associate Professor with the School of Computer, Shenyang Aerospace University, Shenyang, China. Her research interests include computer vision, machine learning, and video content analysis.



**Zhuoqun Fang** received the Ph.D. degree in pattern recognition from Northeastern University, Shenyang, China, in 2019.

He is a Lecturer with the College of Artificial Intelligence, Shenyang Aerospace University, Shenyang. His research interests include reflectance estimation, scene understanding, image processing, and computer vision.



**Lin Zhai** received the B.S. degree from Dalian Ocean University, Dalian, China, in 2011, the M.S. degree from Inner Mongolia Agricultural University, Hohhot, China, in 2014, and the Ph.D. degree from Shandong University, Jinan, China, in 2018.

She is a Lecturer with the School of Science, Shenyang Aerospace University, Shenyang, China. Her research interests include medical image denoising and pattern recognition, especially image denoising.