A New Dataset and Deep Residual Spectral Spatial Network for Hyperspectral Image Classification

Yiming Xue 1, Dan Zeng 1,*, Fansheng Chen 2, Yueming Wang 3, and Zhijiang Zhang 1

1 Key Laboratory of Specialty Fiber Optics and Optical Access Networks, Joint International Research Laboratory of Specialty Fiber Optics and Advanced Communication, Shanghai Institute of Advanced Communication and Data Science, ShangDa road 99, Shanghai 200444, China; xueyiming@shu.edu.cn (Y.X.); zjzhang@shu.edu.cn (Z.Z.)
2 Key Laboratory of Intelligent Infrared Perception, Chinese Academy of Sciences, YuTian road 500, Shanghai 200083, China; cfs@mail.sitp.ac.cn
3 Key Laboratory of Space Active Opto-Electronics Technology, Shanghai Institute of Technical Physics, Chinese Academy of Sciences, YuTian road 500, Shanghai 200083, China; wangym@mail.sitp.ac.cn

* Correspondence: dzeng@shu.edu.cn

Received: 1 March 2020; Accepted: 16 March 2020; Published: 5 April 2020

Abstract: Due to the limited varieties and sizes of existing public hyperspectral image (HSI) datasets, the classification accuracies are higher than 99% with convolutional neural networks (CNNs). In this paper, we presented a new HSI dataset named Shandong Feicheng, whose size and pixel quantity are much larger. It also has a larger intra-class variance and a smaller inter-class variance. State-of-the-art methods were compared on it to verify its diversity. Otherwise, to reduce overfitting caused by the imbalance between high dimension and small quantity of labeled HSI data, existing CNNs for HSI classification are relatively shallow and suffer from low capacity of feature learning. To solve this problem, we proposed an HSI classification framework named deep residual spectral spatial network (DRSSN). By using shortcut connection structure, which is an asymmetry structure, DRSSN can be deeper to extract features with better discrimination. In addition, to alleviate insufficient training caused by unbalanced sample sizes between easily and hard classified samples, we proposed a novel training loss function named sample balanced loss, which allocated weights to the losses of samples according to their prediction confidence. Experimental results on two popular datasets and our proposed dataset showed that our proposed network could provide competitive results compared with state-of-the-art methods.

Keywords: hyperspectral image (HSI) classification; Shandong Feicheng HSI dataset; deep residual spectral spatial network (DRSSN); sample balanced loss

1. Introduction

Hyperspectral image (HSI) consists of hundreds of narrow contiguous wavelength bands carrying a wealth of spectral information. Taking advantage of the rich spectral information, classification using hyperspectral data has been developed for a variety of applications, such as image segmentation, object recognition, land cover mapping and anomaly detection [1–4].

The difficulty in HSI classification lies in the inherent data characteristics of HSI data: First, the high-dimensions of hyperspectral pixels and information redundancy between adjacent bands lead to high calculation cost. Secondly, factors such as different shooting time, different shooting environment or physical limitations of acquisition technology may cause the problem of large intra-class variance and small inter-class variance. As a result, the data structure of HSI is highly nonlinear, which greatly increases the difficulty of classification. Thirdly, the unbalanced category sizes in the HSI dataset often make the training stage more difficult to converge.
Regarding the issues above, many supervised methods and unsupervised methods were proposed [5]. Unsupervised learning methods include [6–8] and commonly used supervised learning methods contain support vector machines (SVMs) and kernel-based methods [9], Bayesian models [10], random forest (RF) [11], and neural networks [12] etc.

However, satisfactory classification results still cannot be obtained by using these methods. Since deep learning has achieved excellent results in many computer vision tasks recently, a series of deep learning based methods have been proposed for HSI classification tasks [13]. These kinds of methods can be regarded as nonlinear mapping from feature space to label space with a hierarchical structure. Using this method, the mapping can be decomposed into a series of nested simple mappings, which has better expression ability. In general, the main methods of solving HSI classification tasks by deep learning can be divided into the following three types:

1. Spectral-based classification methods: Using a convolutional neural network (CNN), the input is a 1-D vector obtained from the spectral band of each pixel. For this type of methods, a stacked autoencoder (SAE) was proposed as a feature extractor to capture the representative stacked spectral and spatial features with a greedy layerwise pretraining strategy. Subsequently, denoising SAE [14], and Laplacian SAE [15] were successively proposed. However, since these models have the requirement that the input must be 1-D data, spatial information is ignored. Besides, there are so many parameters produced by fully connected (FC) layers in these networks that a large number of available samples are required to train the network.

2. Spatial-based classification methods: These methods consider the neighboring pixels of a target pixel in the original remote sensing images in order to extract the spatial feature representation. Therefore, a 2-D CNN architecture is adopted, where the input data is a patch of P×P neighboring pixels. In order to extract high-level spatial features, multi-scale structure methods have been proposed. For example, in [16] the neighboring pixels of each target pixel of the HSI are fed to the network. Compared with SAE, these types of methods use the spatial information to improve the classification performance. However, it should be noted that such methods usually require a pre-processing of the spectral information (such as PCAs [17,18] or autoencoders [19,20]) to reduce the number of bands used for classification, which will lose some of the spectral information.

3. Spectral-spatial classification methods: By using a combination of spatial and spectral information [21], these types of methods can significantly improve the classification accuracy. Each of the target pixel is associated with a P×P spatial neighborhood and B spectral bands (P×P×B). Then they are processed by means of 3-D CNNs in order to learn the local signal changes in both the spatial and the spectral domain of the hyperspectral data cubes. In these types of methods, [22] proposed a 3-D CNN to take full advantage of the structural characteristics of the 3-D hyperspectral remote sensing data.

Although the above deep learning based methods can make full use of both spectral and spatial information, the sample size of training set is limited compared with the dimensionality of HSI data, which usually results in insufficient training and overfitting (also known as Hughes). Besides, the sample size unbalance between easily and hard classified samples prevents the network from being adequately trained.

To solve the information loss [23] caused by gradient vanishing when constructing deep CNNs, the structure of shortcut connection, which is an asymmetry structure, is used in the proposed DRSSN to extract features with better discrimination in a deeper level. We use 2-D convolution to deal with 3-D HSI data with both spectral and spatial information, which can greatly reduce parameter quantity and alleviate overfitting.
Since the sample size unbalance between easily and hard classified samples will cause insufficient network training, a sample balanced loss was proposed to automatically allocate weights for samples based on prediction confidence. During the back propagation of gradient, the loss of easily classified samples will be reduced and the loss of hard classified samples will be increased. In this case, the network can put more emphasis on hard classified samples and further improve classification accuracy.

On the other hand, the classification accuracies of public HSI datasets are over 99% due to their small scale and easily characterized data. For example, the image size of Indian Pines dataset is $145 \times 145$ with 16 categories while the size of Pavia University dataset is $630 \times 340$ with 9 categories. In this case, we presented a new HSI dataset named Shandong Feicheng which has larger scale ($2000 \times 2700$ and $2100 \times 2840$) and more categories (19 categories).

The major contributions of this paper are listed as follows.

1. A new HSI dataset named Shandong Feicheng was presented, which is larger in scale and is more complex in data compared with other public HSI datasets. State-of-the-art methods were tested on the proposed dataset.
2. We proposed a novel HSI classification framework DRSSN. Taking the advantage of the structure of shortcut connection and 2-D convolution, it is much deeper to extract features with better discrimination while reducing overfitting.
3. A novel sample balanced loss was proposed to alleviate insufficient training caused by sample size unbalance between easily and hard classified samples. Experimental results proved its validity.

The remainder of this paper is organized as follows. In Section 2, we describe in detail the proposed Shandong Feicheng and other datasets used in this paper, DRSSN network framework and the sample balanced loss function. Section 3 validates the proposed approach by comparing it with other CNN implementations in the literature. Section 4 discusses the influence of several factors. Section 5 concludes the paper with some remarks and hints at plausible future research lines.

2. Materials and Methods

2.1. Datasets

During our experiment, our proposed Shandong Feicheng dataset and two widely used hyperspectral image datasets, Indian Pines and Pavia University datasets, were adopted to validate the proposed methods.

The Shandong Feicheng dataset presented in this paper was obtained by the new generation of airborne high-resolution imaging spectrometer (high score special aviation hyperspectral spectrometer) in China. It was imaged on two areas in the Feicheng area of Shandong on June 23, 2018. The Shandong Feicheng scene has two images with 63 spectral channels in the 0.4–1.0 $\mu$m region of the visible and infrared spectrum with a spatial resolution of 10 m and a spectral resolution of 12.5 nm. The proposed dataset contains two hyperspectral images, Shandong Downtown and Shandong Suburb. Figure 1 are their false color images. Nineteen land-cover categories were selected and the number of samples for each category is given in Figure 2. It should be noted that these two images are duplicated in some categories.
The major contributions of this paper are listed as follows. We proposed a novel HSI classification framework DRSSN. Taking the advantage of the structure complexity in data compared with other public HSI datasets. State-of-the-art methods were tested on other datasets in the literature. Section 4 discusses the influence of several factors. During our experiment, our proposed Shandong Feicheng dataset and two widely used datasets were selected and the number of samples for each category is given in Figure 1. It should be noted that these two images are duplicated in some categories.

Figure 1. Shandong Feicheng dataset. (a) False color image with three spectral bands of Shandong Downtown; (b) false color image with three spectral bands of Shandong Suburb.

| Color         | Land cover type       | Samples  |
|---------------|-----------------------|----------|
| Background    |                       | 3455537  |
| Trees         |                       | 172506   |
| Shrubs        |                       | 16779    |
| Shadows       |                       | 55997    |
| Polished Tile |                       | 117083   |
| Mosaic Tile   |                       | 151970   |
| Cars          |                       | 12174    |
| Painted Metal Sheets |                 | 53241    |
| Cement Roof  |                       | 102208   |
| Asphalt Roof |                       | 145050   |
| Terracotta Roof |                    | 79773    |
| Bitumen      |                       | 68685    |
| Cement Floor |                       | 86804    |
| Terrazo      |                       | 613042   |
| Cement Track |                       | 242518   |
| Soil         |                       | 30333    |
| Total samples |                      | 5480000  |

Figure 2. Shandong Feicheng dataset (a) Ground-truth image of Shandong Downtown; (b) Number of pixels for each land-cover category of Shandong Downtown; (c) Ground-truth image of Shandong Suburb; (d) Number of pixels for each land-cover category of Shandong Suburb.
When labeling the proposed dataset, to make it has a larger intra-class variance, the sample size of each category is much larger than other public HSI datasets and these samples are widely spread in the images. For example, the average sample size in Pavia University is only 4753, but in our proposed dataset it reaches 175,144. To make the proposed dataset have a smaller inter-class variance, the categories we chose were fine-grained, such as Polished Tile and Mosaic Tile, although they are very similar, we divided them into two separate categories. The intra-class variance and inter-class variance of these datasets are shown in Table 1. It can be seen that the inter-class variance of the proposed Shandong Feicheng dataset is much smaller than the other two datasets. The intra-class variance is basically on the same order of magnitude and the proposed Shandong Downtown has the largest intra-class variance.

| Dataset          | Image Size | Spectral Bands | Classes | Labeled Pixels | Intra-Class Variance | Inter-Class Variance |
|------------------|------------|----------------|---------|----------------|----------------------|----------------------|
| Indian Pines     | 145 × 145  | 220            | 16      | 10,249         | 7.78 × 10^8          | 6.51 × 10^8          |
| Pavia University | 630 × 340  | 103            | 9       | 42,776         | 1.25 × 10^9          | 5.33 × 10^9          |
| Shandong Downtown| 2000 × 2700| 63             | 19      | 3,327,729      | 1.51 × 10^9          | 1.96 × 10^6          |
| Shandong Suburb  | 2100 × 2840| 63             | 19      | 3,327,729      | 8.85 × 10^8          | 1.45 × 10^5          |

As can be seen in Table 1, the Shandong Feicheng dataset proposed in this paper is much larger in size and categories compared with the Indian Pines and Pavia University. The size of Shandong Downtown is 2000 × 2700 which is 256 times and 26 times of the Indian Pines and Pavia University datasets. There are 1,944,463 labeled pixels covering 36% of the entire HSI pixels, which is 189 times and 45 times of these public HSI datasets. The size of Shandong Suburb is 2100 × 2840. It contains a total of 5,964 million pixels and 7 categories. There are 1,383,266 labeled pixels which is 23.19% of the entire HSI. On the whole, the Shandong Feicheng dataset contains 19 categories which covers most of the objects in the images. The size of Shandong Feicheng dataset size is 283 times and 28 times of that in Indian Pines and Pavia University datasets.

Indian Pines is the earliest dataset for HSI classification, which was gathered in 1992 by the airborne visible/infrared imaging spectrometer (AVIRIS) sensor [24] over a set of agricultural fields with regular geometry and with a multiple crops and irregular patches of forest in Northwestern Indiana. The AVIRIS Indian Pines scene has 145 × 145 pixels with 220 spectral channels in the 0.4–2.5 µm region of the visible and infrared spectrum with a spatial resolution of 20 m and a spectral resolution of 10 nm. The number of bands is reduced to 200 by removing water absorption bands. Sixteen different land-cover categories are provided in the ground truth. The number of samples for each category is shown in Figure 3.
Symmetry 2020, 12, x FOR PEER REVIEW 6 of 19

(a) ![Figure 3](image-url) Indian Pines dataset. (a) Representing sixteen land-cover categories; (b) number of pixels for each land-cover category.

Pavia University dataset was collected by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor [25] during a flight campaign over the city of Pavia, in northern Italy. The dataset has 115 spectral bands in range of 0.43 to 0.86 μm with 610 × 340 pixels. The high spatial resolution of 1.3 m per pixel aims to avoid a high percentage of mixed pixels. In the experiment, noisy bands have been removed and the remaining 103 channels were used for classification. Nine land-cover categories were selected and the number of samples for each category is given in Figure 4.

(b) ![Figure 4](image-url) Pavia University dataset. (a) Representing nine land-cover categories; (b) number of pixels for each land-cover category.
2.2. Network Architecture

The DRSSN network structure proposed in this paper is shown in Figure 5. It contains a data input layer, two residual blocks, three fully connected layers and an output layer. Firstly, the proposed DRSSN network structure is explained in detail. Then we introduce the preprocessing strategy for HSI which is used to obtain 3-D input data with both spectral and spatial information.

![Network architecture of deep residual spectral spatial network (DRSSN).](image)

**Figure 5.** Network architecture of deep residual spectral spatial network (DRSSN).

### 2.2.1. Details of DRSSN Framework

Since constructing deep CNNs [26] for HSI classification task is very challenging given the high data dimensionality and the relatively small data quantity, the structure of shortcut connection [27] is added to facilitate the propagation of gradients. Therefore, DRSSN can perform more robustly since deeper architectures can learn deeper feature representation.

As shown in Figure 5, in DRSSN, the 3-D data with size $n \times d \times d$ is first fed to the data input layer, whose structure can be expressed as Conv-BatchNorm-ReLU-MaxPooling, where Conv layer contains $n_1$ convolution kernels of size $n \times k_1 \times k_1$. This module performs a first spectral–spatial feature extraction from the input data, preparing its output feature maps for the rest of the network. It should be noted that the input data is down-sampled in both the convolutional layer and the max pooling layer, so the length and width of the output feature map is $1/4$ times the original data. After that, the obtained feature maps will be fed into two residual blocks in turn. These two residual blocks have a similar structure except the number of convolution layers. As shown in Figure 5, the structure of residual blocks can be divided into two parts: residual mappings and identity mappings. The structure of residual mappings can be expressed as Conv1-BatchNorm-ReLU-Cov2-BatchNorm-ReLU-Cov3-BatchNorm, and the structure of identity mappings can be expressed as Conv-BatchNorm. The output of the residual block is the element-wise sum of both the feature maps output from the residual mappings and the identity mappings. It should be noted that the size of feature maps does not change in these two residual blocks, and only the number of channels is gradually increased in each residual block. The feature map obtained by the two residual blocks will be fed into three fully connected layers to be further integrated. The channel dimensions of the three fully connected layers are $d^{f_3}$, $d^{f_2}$ and $d^{f_1}$ respectively, where $d^{f_3}$ is the category number of the current dataset. To perform classification by utilizing the learned features from the CNNs, we employ a logistic regression classifier, and use Softmax as its output layer activation. Softmax ensures that the activation of each output unit sums to one so that we can deem the output as a set of conditional probabilities. The detailed configuration of the proposed DRSSN is shown in Table 2.
Table 2. Detailed configuration of the DRSSN network: $I_c$ and $O_c$ are the number of input and output channels; $N_b$ and $N_c$ are the number of bands and categories of hyperspectral datasets used for training.

| DRSSN Topologies | Kernel Size ($I_c \times O_c \times L \times L$) | Batch Norm | ReLU | Pooling ($L \times L$) |
|------------------|-----------------------------------------------|-------------|------|------------------------|
| Data Input Layer | C1 64 × 64 × 7 × 7 | Yes | Yes | 3 × 3 |
| Residual Block 1 | C2_1 64 × 64 × 1 × 1 | Yes | Yes | No |
|                  | C2_2 64 × 64 × 3 × 3 | Yes | Yes | No |
|                  | C2_3 64 × 256 × 1 × 1 | Yes | No | No |
|                  | C2_S 64 × 256 × 1 × 1 | Yes | No | No |
| Residual Block 2 | C3_1 256 × 128 × 1 × 1 | Yes | Yes | No |
|                  | C3_2 128 × 128 × 3 × 3 | Yes | Yes | No |
|                  | C3_3 128 × 512 × 1 × 1 | Yes | No | No |
|                  | C3_S 256 × 512 × 1 × 1 | Yes | No | No |
| F1  | $N_f \times 2048$ | Yes | Yes |
| F2  | 2048 × 2048 | Yes | Yes |
| F3  | 1024 × $N_c$ | Yes | Yes |

2.2.2. Data Preprocessing

For many HSI classification methods, the lower right patch of the target pixel is used as 3-D input data to add some spatial information. However, in this paper, we fed the network with a neighborhood patch centered around each pixel with size of $d \times d \times n$. Here $d$ is the patch size and $n$ is the band number of the hyperspectral image. This is a more reasonable design because further the pixels are from the target, the less it contributes to the classification. Moreover, as different from the four-neighbor or eight-neighbor methods [28], to make full use of the spatial information of neighborhood, we choose to extract features from a larger neighborhood. In this paper, $d$ is set to 15–29 in experiments.

If the pixels are near the borders of the image, a zero-padding operation is performed to the excess portion. Compared with discarding these samples or mirror filling them [21], the zero-padding operation is simpler and does not affect the classification accuracy. After data preprocessing, each sample is transformed from 1-D vector into a 3-D input data in $d \times d \times n$ size, which can provide both spectral and spatial information.

2.3. Sample Balanced Loss

In this section, we first illustrate the problem of insufficient training due to unbalanced sample sizes between easily and hard classified samples in HSI datasets. Then we describe the proposed Sample Balanced Loss in detail and explain how it solves the above problem and further improves the classification performance of DRSSN.

$$CE(p_t) = -\log(p_t), (1)$$

In the HSI classification network, the commonly used loss function is cross entropy loss. Its formula is Equation (1), where $p_t$ is the prediction confidence that the samples belong to the target category. During the training process, back propagation algorithm minimizes $CE(p_t)$ to update the network parameters. However, there is often an imbalance between the easily and hard classified samples in HSI datasets. For example, in the Indian Pines dataset, 10 of 16 categories such as the Corn-mintill, Grass-pasture-mowed are easily classified samples whose classification accuracies are
near to 100%, while categories like the Alfalfa and Corn are often hard classified samples. There are two factors that distinguish between hard classified samples and easily classified samples. The first is the complexity of the sample distribution in each category. Table 3 lists the sample size, intra-class variance and the classification accuracy of some categories in the Indian Pines dataset listed in [22]. The author used 20% samples for training, and the rest of them for testing. Among these categories, the samples from Grass-Pasture and Grass-trees categories are easily classified samples, and their classification accuracy is very high, but the classification accuracy of Corn and Soybean-notill categories is worse. The main reason is that compared with easily classified categories, the intra-class variance of hard classified categories is larger. The second influencing factor is the sample size of the category. From Table 3, it can be seen that although the intra-class variance of the Alfalfa and Grass-pasture-mowed categories are small, their classification accuracy is still poor. This is due to the sample size of these categories is small, which will lead to insufficient training of the network for these categories. In this case, during the early stage of training, the easily classified samples are often well-trained and will be assigned to the correct categories with high prediction confidence. On the other hand, the prediction confidence of hard classified samples is relatively low due to insufficient training. Therefore, easily classified samples comprise the majority of loss and dominate the gradient.

Table 3. The sample size, intra-class variance and classification accuracy of some categories in the Indian Pines dataset listed in [22].

| Category         | Samples | Intra-class Variance ($\times10^6$) | Accuracy (%) | Classification Difficulty |
|------------------|---------|------------------------------------|--------------|--------------------------|
| Grass-Pasture    | 483     | 200.84                             | 99.24        | Easily classified samples |
| Grass-trees      | 730     | 150.57                             | 99.51        |                         |
| Corn             | 237     | 2864.47                            | 97.79        |                         |
| Soybean-notill   | 972     | 640.86                             | 98.74        | Hard classified samples |
| Alfalfa          | 46      | 48.96                              | 97.82        |                         |
| Grass-pasture-mowed | 28  | 25.38                              | 98.70        |                         |

In response to this problem, we proposed a novel loss function named Sample Balanced Loss, which can automatically allocate lower loss to easily classified samples and higher loss to hard classified samples. In this way, the network can pay more attention to the hard classified samples during training. The calculation formula of Sample Balanced Loss is shown in Equation (2), where $\log(p_i)^\alpha$ denotes an adjustment factor, and $\alpha \geq 0$ denotes a tunable focusing parameter used to control the attenuation of easily classified samples.

$$SB(p_t) = -[\log(p_t^{-1})]^\alpha \log(p_t),$$  \hspace{1cm} (2)

Comparing Equations (1) and (2) we can see that the cross entropy loss is the same as Sample Balanced Loss with $\alpha = 0$. When the prediction confidence of a sample is greater than 0.5, which means it is an easily classified sample, its cross entropy loss is still high. In this case, the loss of a large amount of easily classified samples is likely to cover up the loss of hard classified samples. However, when the value of $\alpha$ increases, the loss value will decrease rapidly. For example, when $\alpha = 2$, if the prediction probability is 0.9, the loss value obtained by sample balanced loss is only 0.21% of the value obtained by using cross entropy loss, and only 0.0019% when the prediction probability is 0.99. In this case, the loss of the easily classified samples can be reduced greatly, so that the network can pay more attention to hard classified samples and further improve the classification performance of DRSSN.

It should be noted that although the larger $\alpha$ will make the sample balanced loss put higher attenuation to the easily classified samples, it also makes the network unable to further improve the classification performance for them. We can treat OHEM (Online Hard Example Mining) as an extreme situation of large $\alpha$. The purpose of OHEM is to ensure that the training samples are hard training samples. OHEM first sort the samples by loss and perform non-maximum suppression, and then select
N samples with the highest loss for training. The drawback of this method is that it removes all the easily classified samples, which make it difficult for the network to further improve their accuracies. Therefore, the value of $\alpha$ is not the larger the better. In the experiment, we found that we can get the best classification performance with $\alpha = 1$.

3. Results

In this section, we introduced the implementation details, and evaluated the proposed methods using classification metrics, such as overall accuracy (OA), average accuracy (AA), and Kappa coefficient. Among them, OA is the correct accuracy for all test samples. AA is the mean of the classification accuracy of all categories. The Kappa metrics is used to judge whether different models or analysis methods are consistent in predictability. Equation (3) is the calculation formula of Kappa metrics, where $P_o$ is the overall accuracy. Assuming that the number of samples in each category is $a_1, a_2, \ldots, a_C$, the number of predicted samples for each category is $b_1, b_2, \ldots, b_C$ and the number of whole samples is $n$, then the calculation formula for $P_e$ is Equation (4).

$$k = \frac{p_o - p_e}{1 - p_e}$$

$$p_e = \frac{a_1 \times b_1 + a_2 \times b_2 + \ldots + a_C \times b_C}{n \times n}$$

We adopted the Indian Pines, Pavia University and our proposed Shandong Feicheng datasets for assessing the classification performance of the DRSSN. We ran experiments for ten times with randomly selected training data and reported the mean and standard deviation of main classification metrics.

3.1. Implementation Details

In our experiments, the base learning rate is set to 0.01, and the step and maximum iteration period is 20 epochs and 50 epochs. For the stochastic gradient descent (SGD) optimization algorithm, the batch size is set to 100, the weight decay is set to $1 \times 10^{-4}$, and the momentum is set to 0.9. In all experiments, all the filter weights are initialized by Gaussian distribution with zero mean and unit variance.

We use the Python language and Pytorch library to implement the proposed HSI classification network DRSSN. All the implementations were evaluated on the Ubuntu 16.04 operating system with one 3.8 GHz 6-core CPU and 128 GB memory. Additionally, a GTX 1080Ti graphics processing unit (GPU) was used to accelerate computing.

The sample size of categories in training sets is often unbalance in HSI datasets. For example, when using 10%–20% data for training, the OA of many state-of-the-art methods is usually high, but AA is relatively low. The reason is that categories with poor classification performance due to fewer training samples will have a great impact on AA during testing, because AA is the mean of the classification accuracy of all categories. However, they will have little effect on OA because the testing sample size of these categories is small. As a result, we tried to balance the training sample size in accordance with the available sample size for each category.

Therefore, during the experimental stage, HSI samples are divided into training sets and testing sets by the following method. The first step is to randomly divide the original dataset into two subsets: the training set with 75% of the samples and the testing set with the remaining 25%. Then we set a maximum number of samples per category (as a threshold) and reduce the quantity of samples until a balanced result is achieved. For those categories with large sample sizes, we simply decrease the quantity of samples until it has reached the threshold. However, for categories that have very few samples and do not reach the threshold, we use all the available pixels. The maximum sample size per category is set to 200 during the experiments.
3.2. Experimental Results and Analysis

The proposed DRSSN was compared with state-of-the-art HSI classification methods proposed in [21] and [22] on several datasets. In [21] the author proposed an improved 3-D deep CNN model composed by 7 layers which used all the spatial-spectral information of the HSI data. A border managing strategy and a speed-up implementation in graphics processing units (GPUs) were also introduced. Moreover, [22] proposed a supervised residual network using 3-D convolution with consecutive learning blocks that takes the characteristics of HSI into account, it processed the spatial-spectral information in two steps. Although [21] also used the residual method, it is totally different from our proposed DRSSN. The proposed DRSSN used 2-D convolution to process spectral-spatial information at once. The use of 2-D convolution can greatly reduce the network parameters, so DRSSN (with 29 layers) can be deeper than [21] (with 16 layers) to extract features with better discrimination. Besides, since DRSSN processes spectral-spatial information at one time, its structure is simpler, but it can be seen from the experimental results that it could provide competitive results compared with state-of-the-art methods. On the other hand, although [29] also proposed an end-to-end 3-D lightweight CNN for limited neural network and achieved great performance, this method needed other HSI datasets for pre-training, so we just compared our method with CNNs proposed in [21] and [22]. It should be noted that larger input patch size leads to worse performance in [22]. In this case, to make a fair comparison, the results shown in Tables 4 and 5 are based on the same size of training samples.

Table 4. Classification accuracies obtained by different neural networks tested using the Indian Pines dataset: (1) The first column: comparison between the results obtained by convolutional neural network (CNN) in [21] and the results obtained by our deep residual spectral spatial network (DRSSN); (2) The second column: comparison between the results obtained by CNN in [22] and the results obtained by our DRSSN. We repeated the experiment 10 times.

| Indian Pines          | Reference [21] Versus the Proposed DRSSN | Reference [22] Versus the Proposed DRSSN |
|-----------------------|------------------------------------------|------------------------------------------|
|                       | [21] d = 29 Samples | [21] d = 29 Samples | [22] d = 27 Samples | [22] d = 27 Samples |
| Alfalfa               | 99.13 (1.06)          | 100.00 (0.00)         | 97.82               | 97.39 (3.98)          |
| Corn-notill           | 98.17 (0.67)          | 98.94 (1.06)          | 99.17               | 99.42 (0.38)          |
| Corn-mintill          | 98.92 (0.68)          | 99.70 (0.52)          | 99.53               | 99.12 (0.78)          |
| Corn                  | 100.00 (0.00)         | 100.00 (0.00)         | 97.79               | 98.43 (1.84)          |
| Grass-Pasture         | 99.71 (0.21)          | 99.48 (0.90)          | 99.24               | 98.20 (1.59)          |
| Grass-trees           | 99.40 (0.52)          | 99.66 (0.39)          | 99.51               | 99.39 (0.47)          |
| Grass-pasture-mowed   | 100.00 (0.00)         | 100.00 (0.00)         | 98.70               | 98.21 (4.99)          |
| Hay-windrowed         | 100.00 (0.00)         | 100.00 (0.00)         | 99.85               | 99.92 (0.25)          |
| Oats                  | 100.00 (0.00)         | 100.00 (0.00)         | 98.50               | 100.00 (0.00)         |
| Soybean-notill        | 98.62 (1.39)          | 98.97 (0.73)          | 98.74               | 99.22 (0.66)          |
| Soybean-mintill       | 96.15 (0.57)          | 99.49 (0.44)          | 99.30               | 99.78 (0.17)          |
| Soybean-clean         | 99.33 (0.18)          | 100.00 (0.00)         | 98.43               | 98.97 (0.90)          |
| Wheat                 | 99.90 (0.20)          | 100.00 (0.00)         | 100.00              | 99.71 (0.55)          |
| Woods                 | 98.96 (0.46)          | 100.00 (0.00)         | 99.31               | 99.79 (0.25)          |
| Buildings-Grass-Trees | 100.00 (0.00)         | 100.00 (0.00)         | 99.20               | 99.17 (1.15)          |
| Towers                | 100.00 (0.00)         | 97.22 (4.81)          | 97.82               | 98.15 (2.77)          |
| OA (%)                | 98.37 (0.17)          | 99.53 (0.19)          | 99.19 (0.26)        | 99.41 (0.15)          |
| AA (%)                | 99.27 (0.11)          | 99.59 (0.40)          | 98.93 (0.59)        | 99.07 (0.52)          |
| Kappa (%)             | 98.15 (0.19)          | 99.47 (0.22)          | 99.07 (0.30)        | 99.33 (0.17)          |
| Total samples         | 2466                  | 2055                  |                     |                      |
Table 5. Classification accuracies obtained by different neural networks tested using the Pavia University dataset: (1) The first column: comparison between the results obtained by CNN in [21] and the results obtained by our DRSSN; (2) The second column: comparison between the results obtained by CNN in [22] and the results obtained by our DRSSN. We repeated the experiment 10 times.

| Pavia University | Reference [21] Versus the Proposed DRSSN | Reference [22] Versus the Proposed DRSSN |
|------------------|------------------------------------------|------------------------------------------|
|                  | [21] d = 27 Samples                      | [22] d = 27 Samples                      |
| Asphalt          | 96.31 (0.19) 97.08 (1.17) 200            | 99.92 99.88 (0.08) 664                  |
| Meadows          | 97.54 (0.39) 99.28 (0.18) 200            | 99.96 99.97 (0.02) 1865                 |
| Gravel           | 96.84 (0.29) 99.66 (0.60) 200            | 98.46 99.47 (0.50) 210                  |
| Trees            | 97.58 (0.41) 98.69 (0.60) 200            | 99.69 98.84 (0.26) 307                  |
| Painted metal sheets | 99.65 (0.15) 99.94 (0.12) 200 | 99.99 99.79 (0.14) 135 |
| Bare Soil        | 99.33 (0.25) 99.78 (0.22) 200            | 99.94 100.00 (0.00) 503                  |
| Bitumen          | 98.90 (1.14) 100.00 (0.00) 200            | 99.82 99.62 (0.30) 133                  |
| Self-Blocking    | 98.89 (0.47) 99.02 (0.39) 200            | 99.22 99.76 (0.14) 369                  |
| Bricks           | 99.58 (0.09) 99.49 (0.32) 200            | 99.95 99.00 (0.73) 95                   |
| OA (%)           | 97.80 (0.22) 99.00 (0.16) 1800           | 99.79 (0.09) 99.80 (0.05) 4281         |
| AA (%)           | 98.29 (0.25) 99.22 (0.12) 1800           | 99.66 (0.17) 99.59 (0.12) 4281         |
| Kappa (%)        | 97.44 (0.29) 98.68 (0.21) 1800           | 99.72 (0.12) 99.74 (0.06) 4281         |

Table 4 shows detailed comparison between different tested neural networks on Indian Pines dataset. In the same training environment, the proposed DRSSN is +1.16%, +0.32% and +1.32% for OA, AA and Kappa metrics compared with the CNN described in [21]. Compared with [22], OA is increased by 0.22%, AA is increased by 0.14% and Kappa metrics is increased by 0.26%.

Table 5 shows detailed comparison between the different tested neural networks using Pavia University dataset. Compared with [21], the proposed DRSSN is +1.20%, +0.93% and +1.24% for OA, AA and Kappa metrics, respectively. Compared with [22], the result of DRSSN has a small lead in OA and Kappa metrics, and AA is slightly decreased.

According to the reported results, the deep features with the proposed DRSSN achieved higher classification performance than other state-of-the-art methods. In addition, although the same training sample size in [21] and [22] were used for comparison experiments, we consider that using a fixed maximum number of training samples for each category is a more reasonable training method, because the sample sizes of categories in HSI datasets are often unbalanced. If using 10%–20% of the data to train, they will still be unbalanced. For example, if we take 10% of the data in the Pavia University dataset for training, the Gravel category will only have 209 samples, while the Meadows category still has 1864 samples. However, if the maximum number of training samples for each category is fixed, both the categories will have 200 samples. In this case, the unbalance sample sizes between categories will be reduced. Moreover, in the HSI classification task, the collection of labeled samples is often costly, so how to make the network achieve good classification performance on the basis of using less samples is also a subject worth studying, and this strategy can help us to control the sample size of categories easily.

On the other hand, we also evaluated the classification performance of DRSSN and CNNs in [21], [22] on the proposed Shandong Feicheng dataset. In the experiment, the maximum sample size threshold for each category was set to 200. From the above experiments, we can see that most methods have achieved an approximate saturation accuracy, which is up to 99.00% on the Indian Pines and Pavia University datasets. However, from Tables 6 and 7, it can be seen that there is a big drop in classification accuracies on the two HSI image of the Shandong Feicheng dataset. The main reason is that the scale of Shandong Feicheng dataset is much larger, which makes its distribution of features more complex. Therefore, it is more difficult to train a robust classification network with 200 samples.
per category on the Shandong Feicheng dataset. Besides, our proposed DRSSN achieved the highest classification accuracy on the Shandong Feicheng dataset, which proves the validity of our framework again. The results of DRSSN can be used as a benchmark for Shandong Feicheng dataset.

Table 6. Classification accuracies obtained by different neural networks tested using the Shandong Downtown dataset. Comparison between the results obtained by CNN in [21,22] and the results obtained by our DRSSN. We repeated the experiment 10 times.

| Category          | [21]  | [22]  | DRSSN  | Samples |
|-------------------|-------|-------|--------|---------|
| Trees             | 84.71 (3.55) | 87.75 (9.74) | 96.28 (1.55) | 200     |
| Shrubs            | 83.83 (7.12) | 43.73 (4.56) | 96.65 (1.33) | 200     |
| Shadows           | 87.38 (1.07) | 96.82 (2.30) | 97.78 (1.37) | 200     |
| Polished-Tile     | 96.09 (1.69) | 98.64 (0.31) | 98.94 (0.35) | 200     |
| Mosaic-Tile       | 84.11 (7.09) | 98.01 (0.75) | 98.41 (0.30) | 200     |
| Cars              | 78.43 (4.70) | 84.24 (4.44) | 97.38 (0.51) | 200     |
| Painted-Metal-Sheets | 98.42 (1.14) | 99.83 (0.08) | 99.87 (0.12) | 200     |
| Cement-Roof      | 47.22 (1.42) | 86.99 (3.72) | 96.84 (0.54) | 200     |
| Asphalt-Roof     | 87.07 (3.41) | 89.57 (2.91) | 97.74 (0.67) | 200     |
| Terracotta-Roof  | 92.02 (1.77) | 92.90 (0.51) | 99.15 (0.43) | 200     |
| Bitumen          | 98.62 (0.59) | 99.61 (0.17) | 99.68 (0.10) | 200     |
| Cement-Floor     | 93.13 (1.50) | 97.21 (0.62) | 96.89 (0.93) | 200     |
| Terrazzo         | 75.97 (6.72) | 78.00 (6.74) | 85.33 (1.57) | 200     |
| Cement-Track     | 62.71 (8.37) | 73.03 (5.42) | 82.34 (1.30) | 200     |
| Soil             | 93.33 (3.98) | 99.29 (0.53) | 99.06 (0.92) | 200     |
| OA (%)           | 79.76 (1.21) | 85.82 (2.07) | 92.02 (0.45) | 200     |
| AA (%)           | 84.20 (2.85) | 88.37 (2.13) | 96.16 (0.21) | 200     |
| Kappa (%)        | 77.20 (1.66) | 83.92 (2.24) | 90.84 (0.50) | 200     |
| Total samples    | 3000   | 3000   | 3000   | 3000    |

Table 7. Classification accuracies obtained by different neural networks tested using the Shandong Suburb dataset. Comparison between the results obtained by CNN in [21,22] and the results obtained by our DRSSN. We repeated the experiment 10 times.

| Category          | [21]  | [22]  | DRSSN  | Samples |
|-------------------|-------|-------|--------|---------|
| Trees             | 78.79 (9.78) | 62.08 (7.32) | 83.13 (1.89) | 200     |
| Shrubs            | 34.81 (5.26) | 37.35 (1.48) | 84.26 (1.90) | 200     |
| Cement-Floor     | 96.85 (0.63) | 98.84 (0.93) | 99.65 (0.19) | 200     |
| Water             | 98.71 (1.31) | 99.83 (0.10) | 99.72 (0.19) | 200     |
| Solar-Panel       | 96.64 (1.89) | 99.80 (0.24) | 99.85 (0.11) | 200     |
| Greenhouse        | 90.89 (3.09) | 96.05 (0.89) | 99.67 (0.11) | 200     |
| Grow-Plants      | 94.84 (2.33) | 93.55 (1.41) | 99.53 (0.18) | 200     |
| OA (%)           | 90.32 (2.77) | 89.13 (1.07) | 95.95 (0.37) | 200     |
| AA (%)           | 84.50 (1.78) | 83.96 (1.29) | 95.15 (0.42) | 200     |
| Kappa (%)        | 88.17 (3.28) | 86.71 (1.28) | 94.99 (0.45) | 200     |
| Total samples    | 1400   | 1400   | 1400   | 1400    |

3.3. Ablation Study

In order to illustrate the contribution of the proposed DRSSN further, we evaluate the contribution of the shortcut connection structure. We compared the classification results obtained by SNN which has the same depth and similar parameter size with DRSSN. In addition, we designed another two DRSSN with one and three residual blocks to observe the influence of the number of residual blocks. To evaluate our proposed loss function used in DRSSN, we performed ablation experiments with cross entropy loss and sample balanced loss on the Pavia University dataset. We also compared the
classification results of DRSSN with and without dropout, which are used to verify its effects on performance. In all the experiments, we used the method described above to divide the HSI datasets into training sets and testing sets, and the maximum number of samples per category was set to 200. In addition, the window size was set to 27 and the sample balanced loss focus parameter $\alpha$ was set to 1 to reduce the influences of other factors on network classification performance.

In the experiment, we simply removed two identity mappings in DRSSN to design SSN. Table 8 is the detailed configuration of SSN. We can observe that compared to the proposed DRSSN, SSN has the same depth and a similar amount of parameters. In this case, we can focus on analyzing the influence of shortcut connection structure. Table 9 shows the classification results of SSN and DRSSN on the Pavia University dataset. It can be seen that DRSSN increases the OA by 0.15%, AA by 0.18% and Kappa metrics by 0.20%. We think the main reason is that the use of shortcut connection makes the gradient pass more smoothly, so with the same depth and similar parameter amount, DRSSN can achieve a better classification performance than SSN.

| SSN Topologies |
|----------------|
| **Kernel Size** $(I_c \times O_c \times L \times L)$ | **Batch Norm** | **ReLU** | **Pooling** $(L \times L)$ |
| Data Input Layer | C1 | $N_b \times 64 \times 7 \times 7$ | Yes | Yes | 3 × 3 |
| C2_1 | 64 × 64 × 1 × 1 | Yes | Yes | No |
| Residual Block 1 | C2_2 | 64 × 64 × 3 × 3 | Yes | Yes | No |
| C2_3 | 64 × 256 × 1 × 1 | Yes | No | No |
| C3_1 | 256 × 128 × 1 × 1 | Yes | Yes | No |
| Residual Block 2 | C3_2 | 128 × 128 × 3 × 3 | Yes | Yes | No |
| C3_3 | 128 × 512 × 1 × 1 | Yes | No | No |

| Pavia University |
|------------------|
| **Class** | **SSN** | **DRSSN** |
| Asphalt | 96.58 (1.18) | 97.08 (1.17) |
| Meadows | 99.22 (0.17) | 99.28 (0.18) |
| Gravel | 99.31 (0.79) | 99.66 (0.60) |
| Trees | 98.46 (0.45) | 98.69 (0.60) |
| Painted metal sheets | 99.82 (0.36) | 99.94 (0.12) |
| Bare Soil | 99.95 (0.06) | 99.78 (0.22) |
| Bitumen | 99.76 (0.23) | 100.00 (0.00) |
| Self-Blocking Bricks | 98.80 (0.59) | 99.02 (0.39) |
| Shadows | 99.41 (0.43) | 99.49 (0.32) |
| OA (%) | 98.85 (0.15) | 99.00 (0.16) |
| AA (%) | 99.04 (0.14) | 99.22 (0.12) |
| Kappa (%) | 98.48 (0.20) | 98.68 (0.21) |
To evaluate the influence of the number of residual blocks, we designed another two DRSSN with one and three residual blocks. As can be seen in Table 10, our proposed DRSSN with two residual blocks can achieve the best classification performance. This indicates that although deep CNNs can extract features with better discrimination, it does not mean the deeper layers the better. In our experiment, we found that setting the number of residual blocks to two is the best choice.

Table 10. Classification performance of DRSSN with different number of residual blocks using the Pavia University dataset. We repeated the experiment 10 times.

| Number of Residual Blocks | One (OA) (%) | Two (OA) (%) | Three (OA) (%) |
|---------------------------|-------------|-------------|---------------|
|                            | 98.66 (0.21)| 99.00 (0.16)| 98.86 (0.18)  |
| AA (%)                    | 99.14 (0.07)| 99.22 (0.12)| 99.02 (0.15)  |
| Kappa (%)                 | 98.23 (0.28)| 98.68 (0.21)| 98.50 (0.24)  |

Table 11 shows the classification performance of DRSSN with cross entropy loss and sample balanced loss on the Pavia University dataset. Using the proposed sample balanced loss, OA is increased by 0.27%, AA is increased by 0.22% and Kappa metrics is increased by 0.35%. This indicates that our proposed sample balanced loss is valid to the problem of insufficient training caused by unbalanced sample sizes between easily classified samples and hard classified samples and can further improve the classification performance of CNN.

Table 11. Classification performance with cross entropy loss or sample balanced loss using the Pavia University dataset. We repeated the experiment 10 times.

| Pavia University            | Cross Entropy Loss | Sample Balanced Loss |
|-----------------------------|--------------------|---------------------|
| Asphalt                    | 95.88 (1.06)       | 97.08 (1.17)        |
| Meadows                    | 99.10 (0.33)       | 99.28 (0.18)        |
| Gravel                     | 99.47 (0.37)       | 99.66 (0.60)        |
| Trees                      | 98.85 (0.46)       | 98.69 (0.60)        |
| Painted metal sheets       | 99.58 (0.45)       | 99.94 (0.12)        |
| Bare Soil                  | 99.79 (0.15)       | 99.78 (0.22)        |
| Bitumen                    | 99.82 (0.15)       | 100.00 (0.00)       |
| Self-Blocking Bricks       | 99.17 (0.50)       | 99.02 (0.39)        |
| Shadows                    | 99.32 (0.57)       | 99.49 (0.32)        |
| OA (%)                     | 98.73 (0.22)       | 99.00 (0.16)        |
| AA (%)                     | 99.00 (0.21)       | 99.22 (0.12)        |
| Kappa (%)                  | 98.33 (0.29)       | 98.68 (0.21)        |

The imbalance between high dimensionality and limited availability of training samples for HSI classification often leads to insufficient training and overfitting (or Hughes). To solve this problem, we added dropout to the last three fully connected layers. Dropout is a method to handle overfitting. It sets the output of some hidden neurons to zero, which means that the dropped neurons do not contribute in the forward pass and they are not used in back propagation. In different training epochs, the deep CNNs form a different neural network by dropping neurons randomly. The dropout method prevents complex co-adaptations. Table 12 shows the classification performance with and without dropout on Pavia University dataset. The classification accuracy is improved after using dropout. In DRSSN, the rate of dropout is set to 0.2.
Table 12. Classification performance with and without Dropout using the Pavia University dataset. We repeated the experiment 10 times.

| Class                     | Without Dropout | With Dropout |
|---------------------------|-----------------|--------------|
| Asphalt                   | 96.45 (1.63)    | 97.08 (1.17) |
| Meadows                   | 99.09 (0.30)    | 99.28 (0.18) |
| Gravel                    | 99.47 (0.37)    | 99.66 (0.60) |
| Trees                     | 98.15 (0.83)    | 98.69 (0.60) |
| Painted metal sheets      | 99.88 (0.15)    | 99.94 (0.12) |
| Bare Soil                 | 99.67 (0.48)    | 99.78 (0.22) |
| Bitumen                   | 99.82 (0.36)    | 100.00 (0.00) |
| Self-Blocking Bricks      | 98.52 (0.56)    | 99.02 (0.39) |
| Shadows                   | 99.07 (0.62)    | 99.49 (0.32) |
| OA (%)                    | 98.70 (0.21)    | 99.00 (0.16) |
| AA (%)                    | 98.90 (0.16)    | 99.22 (0.12) |
| Kappa (%)                 | 98.28 (0.27)    | 98.68 (0.21) |

4. Discussion

In this section, we investigated the effects of important parameters introduced in our method on classification performance.

4.1. The Effect of Window Size

The window size of the target pixel will affect the final classification performance, since the larger window size can contain more spatial and spectral information. However, farther the pixel is from the target pixel, the lower correlation will be with the target pixel. Therefore, it will only increase the cost of calculation. So, experiment is needed to balance the running time and accuracy and get a suitable window size. Here we used \(d\) to denote the window size. We tested different window sizes, using a fixed number of 200 samples per category. We have considered four window sizes: \(15 \times 15\), \(21 \times 21\), \(27 \times 27\) and \(33 \times 33\), which has a fixed growth step size of 6. In Table 13 we illustrated the classification performance of using different window sizes in the Pavia University dataset. It can be observed that as the window size increases, so does the accuracy. However, the required time for each epoch is growing much faster than the accuracy gain achieved by the increased window size. When the window size is increased from 27 to 33, the accuracy gain is very small. So, in terms of the accuracy/time ratio, we chose \(d = 27\) in this paper, although further increasing the window size can achieve better accuracy, its execution time is much larger compared to the small amount of accuracy gain. However, for practical applications, the window size can be adjusted according to the requirements of speed and accuracy.
Table 13. Classification performance of different window-size using the Pavia University dataset. We repeated the experiment 10 times.

| Pavia University | Window Size |
|------------------|-------------|
| Class            | 15 x 15     | 21 x 21     | 27 x 27     | 33 x 33     |
| Asphalt          | 92.96 (1.36)| 95.57 (1.26)| 97.08 (1.17)| 98.13 (1.19)|
| Meadows          | 97.24 (0.30)| 98.96 (0.31)| 99.28 (0.18)| 99.29 (0.29)|
| Gravel           | 95.61 (1.37)| 98.44 (0.77)| 99.66 (0.60)| 98.66 (0.78)|
| Trees            | 98.72 (0.33)| 98.72 (0.34)| 98.69 (0.60)| 98.38 (0.54)|
| Painted metal sheets | 99.82 (0.24)| 99.94 (0.12)| 99.94 (0.12)| 100.00 (0.14)|
| Bare Soil        | 98.42 (1.02)| 99.89 (0.19)| 99.78 (0.22)| 99.88 (0.23)|
| Bitumen          | 98.19 (0.66)| 99.40 (0.54)| 100.00 (0.00)| 100.00 (0.00)|
| Self-Blocking Bricks | 95.39 (1.29)| 98.80 (0.58)| 99.02 (0.39)| 98.48 (0.29)|
| Shadows           | 99.92 (0.17)| 99.49 (0.32)| 99.49 (0.32)| 99.15 (0.25)|
| OA (%)           | 96.75 (0.32)| 98.54 (0.31)| 99.00 (0.16)| 99.06 (0.23)|
| AA (%)           | 97.36 (0.33)| 98.80 (0.19)| 99.22 (0.12)| 99.26 (0.14)|
| Kappa (%)        | 95.74 (0.42)| 98.08 (0.41)| 98.68 (0.21)| 98.76 (0.32)|
| Running Time(s/epoch) | 1.27 (0.02)| 1.55 (0.01)| 2.49 (0.01)| 5.37 (0.02)|

4.2. The Effect of Focusing Parameter

Equation (2) is the calculation formula of sample balanced loss, where \( \log(p_f^{-1})^a \) is an adjustment factor, \( a \geq 0 \) is a tunable focus parameter, which is used to control the degree of attenuation of the easily classified samples. When \( a \) is getting larger, the loss of the easily classified samples will be smaller, so their ability to update network will be weaker. A natural judgment is to increase the value of \( a \) as much as possible, since the easily classified samples have been trained sufficiently. So, suppressing them can help the network to pay more attention to hard classified samples. However, from Table 14 we can see that the classification performance of the network is not improved with the increase of \( a \) from 0.5 to 2.5. It achieved the best result when \( a = 1 \) and started to decrease after further increasing the value of focusing parameter. We believe that the main reason is that as the value of \( a \) is increased, the easily classified samples will contribute less and less to the network. Although this will force the network to pay more attention to the hard classified samples, it prevents the network from further optimizing its performance on easily classified samples. Therefore, an appropriate value of \( a \) is required so that the network can easily maintain the accuracy of easily classified samples and focus more on the hard classified samples. In the experiment, we can see that the optimal value of \( a \) is 1.
Table 14. Classification performance of different focusing parameter using the Pavia University dataset. We repeated the experiment 10 times.

| Class                  | 0.5     | 1       | 1.5      | 2       | 2.5     |
|------------------------|---------|---------|----------|---------|---------|
| Asphalt                | 96.75(1.26) | 97.08(1.17) | 96.85(1.34) | 96.46(1.43) | 98.82(0.38) |
| Meadows                | 99.15(0.17) | 99.28(0.18)   | 98.96(0.49)  | 98.94(0.30)   | 98.82(0.38)  |
| Gravel                 | 99.47(0.56) | 99.66(0.60)   | 99.27(0.31)  | 99.43(0.48)   | 99.16(0.43)  |
| Trees                  | 98.49(0.51) | 98.69(0.60)   | 98.49(0.59)  | 98.20(0.68)   | 98.07(0.43)  |
| Painted metal sheets   | 99.70(0.33) | 99.94(0.12)   | 99.88(0.15)  | 99.88(0.15)   | 99.82(0.24)  |
| Bare Soil              | 99.94(0.08) | 99.78(0.22)   | 99.89(0.19)  | 99.86(0.17)   | 99.59(0.44)  |
| Bitumen                | 100.00(0.00) | 100.00(0.00)  | 99.70(0.47)  | 99.70(0.33)   | 99.58(0.45)  |
| Self-Blocking Bricks   | 99.17(0.27) | 99.02(0.39)   | 98.89(0.39)  | 98.93(0.64)   | 98.87(0.70)  |
| Shadows                | 99.41(0.51) | 99.49(0.32)   | 99.83(0.34)  | 99.66(0.32)   | 99.58(0.38)  |
| OA (%)                 | 98.89(0.20) | **99.00(0.16)** | 98.79(0.25)  | 98.70(0.24)   | 98.51(0.25)  |
| AA (%)                 | 99.12(0.13) | **99.22(0.12)** | 99.08(0.17)  | 99.01(0.19)   | 98.83(0.16)  |
| Kappa (%)              | 98.53(0.26) | **98.68(0.21)** | 98.40(0.33)  | 98.29(0.31)   | 98.04(0.33)  |

5. Conclusions

HSI classification is the key for hyperspectral image analysis. The classification accuracies on public hyperspectral image (HSI) datasets are higher than 99%. The main reason is the limited varieties and sizes of public datasets. We proposed a new HSI dataset named Shandong Feicheng, which has large scale and high data complexity. The declined accuracies of state-of-the-arts on the proposed dataset validated its diversity. In addition, we presented a novel HSI classification framework named DRSSN to manage high dimension and training sample-reduced HSI data. The structure of shortcut connection was added in the proposed network to learn deeper feature representation. A novel sample balanced loss was proposed to solve the problem of insufficient training caused by unbalanced sample sizes between easily and hard classified samples. The network can pay more attention to hard classified samples by automatically allocating higher loss weights to the hard classified samples and lower loss weights to the easily classified ones. The reported results on two public datasets and our proposed Shandong Feicheng datasets demonstrated the effectiveness of our proposed DRSSN which achieved better classification performance than other state-of-the-art methods. Our future work will focus on how to use smaller training sample size to achieve similar or better classification performance.

Author Contributions: Conceptualization, Y.X. and D.Z.; methodology, Y.X.; software, Y.X.; validation, Y.X., D.Z. and Z.Z.; formal analysis, Y.W.; investigation, Y.X.; resources, F.C.; data curation, Y.W.; writing—original draft preparation, Y.X.; writing—review and editing, D.Z.; visualization, Y.X.; supervision, D.Z.; project administration, D.Z.; funding acquisition, D.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number 61572307.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (No. 61572307).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, L.; Zhang, L.; Du, B. Deep Learning for Remote Sensing Data: A Technical Tutorial on the State of the Art. *IEEE Geosci. Remote Sens. Mag.* 2016, 4, 22–40. [CrossRef]

2. Zhang, L.; Zhang, L.; Tao, D.; Huang, X.; Du, B. Hyperspectral Remote Sensing Image Subpixel Target Detection Based on Supervised Metric Learning. *IEEE Trans. Geosci. Remote Sens.* 2014, 52, 4955–4965. [CrossRef]
3. Gevaert, C.M.; Suomalainen, J.; Tang, J.; Kooistra, L. Generation of spectral–temporal response surfaces by combining multispectral satellite and hyperspectral UAV imagery for precision agriculture applications. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 2015, 8, 3140–3146. [CrossRef]

4. Makki, I.; Younes, R.; Francis, C.; Bianchi, T.; Zucchetti, M. A survey of landmine detection using hyperspectral imaging. *ISPRS J. Photogramm. Remote Sens.* 2017, 124, 40–53. [CrossRef]

5. Ghamisi, P.; Plaza, J.; Chen, Y.; Li, J.; Plaza, A.J. Advanced Spectral Classifiers for Hyperspectral Images: A review. *IEEE Geosci. Remote Sens. Mag.* 2017, 5, 8–32. [CrossRef]

6. Haut, J.M.; Paioletti, M.; Plaza, J.; Plaza, A. Cloud implementation of the K-means algorithm for hyperspectral image analysis. *J. Supercomput.* 2017, 73, 1–16. [CrossRef]

7. Marinoni, A.; Gamba, P. Unsupervised Data Driven Feature Extraction by Means of Mutual Information Maximization. *IEEE Trans. Comput. Imaging* 2017, 3, 243–253. [CrossRef]

8. Marinoni, A.; Iannelli, G.C.; Gamba, P. An Information Theory-Based Scheme for Efficient Classification of Remote Sensing Data. *IEEE Trans. Geosci. Remote Sens.* 2017, 55, 5864–5876. [CrossRef]

9. Camps-Valls, G.; Bruzzone, L. Kernel-based methods for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 2005, 43, 1351–1362. [CrossRef]

10. Bazi, Y.; Melgani, F. Gaussian Process Approach to Remote Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* 2010, 48, 186–197. [CrossRef]

11. Ham, J.; Chen, Y.; Crawford, M.M.; Ghosh, J. Investigation of the random forest framework for classification of hyperspectral data. *IEEE Trans. Geosci. Remote Sens.* 2005, 43, 492–501. [CrossRef]

12. Chen, Y.; Lin, Z.; Zhao, X.; Wang, G.; Gu, Y. Deep Learning-Based Classification of Hyperspectral Data. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* 2014, 7, 2094–2107. [CrossRef]

13. He, L.; Li, J.; Liu, C.; Li, S. Recent Advances on Spectral–Spatial Hyperspectral Image Classification: An Overview and New Guidelines. *IEEE Trans. Geosci. Remote Sens.* 2018, 56, 1579–1597. [CrossRef]

14. Vincent, P.; Larochelle, H.; Lajoie, I.; Bengio, Y.; Manzagol, P.A. Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *J. Mach. Learn. Res.* 2010, 11, 3371–3408.

15. Jia, K.; Lin, S.; Gao, S.; Zhan, S.; Shi, B.E. Laplacian Auto-Encoders: An explicit learning of nonlinear data manifold. *Neurocomputing* 2015, 160, 250–260. [CrossRef]

16. Zhong, Z.; Li, J.; Ma, L.; Jiang, H.; Zhao, H. Deep residual networks for hyperspectral image classification. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 1824–1827.

17. Kang, X.; Xiang, X.; Li, S.; Benediktsson, J.A. PCA-based edge-preserving features for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* 2017, 55, 7140–7151. [CrossRef]

18. Agarwal, A.; El-Ghazawi, T.; El-Askary, H.; Le-Moigne, J. Efficient hierarchical-PCA dimension reduction for hyperspectral imagery. In Proceedings of the 2007 IEEE International Symposium on Signal Processing and Information Technology, Giza, Egypt, 15–18 December 2007; pp. 353–356.

19. Villa, A.; Benediktsson, J.A.; Chanussot, J.; Jutten, C. Hyperspectral image classification with independent component discriminant analysis. *IEEE Trans. Geosci. Remote Sens.* 2011, 49, 4865–4876. [CrossRef]

20. Wang, J.; Chang, C.-I. Independent component analysis-based dimensionality reduction with applications in hyperspectral image analysis. *IEEE Trans. Geosci. Remote Sens.* 2006, 44, 1586–1600. [CrossRef]

21. Paioletti, M.E.; Haut, J.M.; Plaza, J.; Plaza, A. A new deep convolutional neural network for fast hyperspectral image classification. *ISPRS J. Photogramm. Remote Sens.* 2018, 145, 120–147. [CrossRef]

22. Zhong, Z.; Member, S.; Li, J.; Member, S.; Luo, Z.; Chapman, M. Spectral–Spatial Residual Network for Hyperspectral Image Classification: A 3-D Deep Learning Framework. *IEEE Trans. Geosci. Remote Sens.* 2018, 56, 847–858. [CrossRef]

23. Gundogdu, E.; Koç, A.; Alatan, A.A. Infrared object classification using decision tree based deep neural networks. In Proceedings of the 2016 24th Signal Processing and Communication Application Conference (SIU), Zonguldak, Turkey, 16–19 May 2016; pp. 1913–1916.

24. Green, R.O.; Eastwood, M.L.; Sarture, C.M.; Chrien, T.G.; Aronsson, M.; Chippendale, B.J.; Faust, J.A.; Pavri, B.E.; Chovit, C.J.; Solis, M.; et al. Imaging Spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS). *Remote Sens. Environ.* 1998, 65, 227–248. [CrossRef]
25. Kunkel, B.; Blechinger, F.; Lutz, R.; Doerrfer, R.; van der Piepen, H.; Schroder, M. ROSIS (Reflective Optics System Imaging Spectrometer)—A Candidate Instrument for Polar Platform Missions. In Optoelectronic Technologies for Remote Sensing from Space; Seeley, J., Bowyer, S., Eds.; International Society for Optics and Photonics: Bellingham, WA, USA, 1988; Volume 0868, pp. 134–141.

26. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.

27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition; in CVPR. 2016. Available online: http://openaccess.thecvf.com/content_cvpr_2016/html/He_Deep_Residual_Learning_CVPR_2016_paper.html (accessed on 28 February 2020).

28. Chen, Y.; Jiang, H.; Li, C.; Jia, X.; Ghamisi, P. Deep Feature Extraction and Classification of Hyperspectral Images Based on Convolutional Neural Networks. IEEE Trans. Geosci. Remote Sens. 2016, 54, 6232–6251. [CrossRef]

29. Zhang, H.; Li, Y.; Jiang, Y.; Wang, P.; Shen, Q.; Shen, C. Hyperspectral Classification Based on Lightweight 3-D-CNN With Transfer Learning. IEEE Trans. Geosci. Remote Sens. 2019, 57, 5813–5828. [CrossRef]

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).