




## Article

# Hyperspectral Image Classification Based on 3D Asymmetric Inception Network with Data Fusion Transfer Learning

Bei Fang <sup>1,†</sup> , Yu Liu <sup>2,†</sup>, Haokui Zhang <sup>3\*</sup> and Juhou He <sup>1,\*</sup>

<sup>1</sup> Key Laboratory of Modern Teaching Technology, Ministry of Education, Shaanxi Normal University, Xi'an 710062, China; beifang@snnu.edu.cn

<sup>2</sup> ByteDance, Singapore 148957, Singapore; unilau13@gmail.com

<sup>3</sup> Intellifusion, Shenzhen 518000, China; hkzhang1991@mail.nwpu.edu.cn

\* Correspondence: juhoh@snnu.edu.cn

† These authors contributed equally to this work.

**Abstract:** Hyperspectral image (HSI) classification has been marked by exceptional progress in recent years. Much of this progress has come from advances in convolutional neural networks (CNNs). Different from the RGB images, HSI images are captured by various remote sensors with different spectral configurations. Moreover, each HSI dataset only contains very limited training samples and thus the model is prone to overfitting when using deep CNNs. In this paper, we first propose a 3D asymmetric inception network, AINet, to overcome the overfitting problem. With the emphasis on spectral signatures over spatial contexts of HSI data, the 3D convolution layer of AINet is replaced with two asymmetric inception units, i.e., a space inception unit and spectrum inception unit, to convey and classify the features effectively. In addition, we exploited a data-fusion transfer learning strategy to improve model initialization and classification performance. Extensive experiments show that the proposed approach outperforms all of the state-of-the-art methods via several HSI benchmarks, including Pavia University, Indian Pines and Kennedy Space Center (KSC).

**Keywords:** hyperspectral image classification; convolutional neural network; light-weight network; 3D asymmetric inception network; transfer learning



**Citation:** Fang, B.; Liu, Y.; Zhang, H.; He, J. Hyperspectral Image Classification Based on 3D Asymmetric Inception Network with Data Fusion Transfer Learning. *Remote Sens.* **2022**, *14*, 1711. <https://doi.org/10.3390/rs14071711>

Academic Editor: Weasley Yuan

Received: 21 February 2022

Accepted: 31 March 2022

Published: 1 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 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 (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Hyperspectral image (HSI) classification is an important research problem in remote sensing (RS) and has a broad range of applications. Differing from RGB images, hyperspectral data is composed of spectral signatures and spatial contexts. On the one hand, it provides abundant spectral-spatial information for “over-band” classification. On the other hand, it raises challenges in extracting high-dimensional features [1–4].

The early HSI classification methods mainly focus on selecting or extracting spectral features due to abundant spectral information derived from the hundreds of contiguous spectral bands. Feature selection (also known as band selection) methods try to find the most representative features (bands) from raw HSI data to preserve their physical meaning. For instance, Wang et al. [5] used manifold ranking as an unsupervised feature-selection method to choose the most representative bands for training the following classifiers. Yin et al. [6] introduced a computational evolutionary strategy into the field of supervised band selection, where the candidate band combinations are evaluated through an affinity function driven by hyperspectral classification accuracy. Feature extraction approaches usually learn representative features through linear or nonlinear transformation. For instance, Huang et al. [7] extended the k-nearest neighbor technique and proposed a feature extraction method called double nearest proportion feature extraction to reduce the dimensionality. Based on linear transformation nonparametric weighted feature extraction (NWFE), Kuo et al. [8] proposed kernel-based NWFE, which has the advantages of both linear and nonlinear transformation.

These spectrum-based approaches select or extract the features directly from the pixel-wise spectra while ignoring the intrinsic geographical structure in HSI data. Recent studies have shown that the combined use of spectral and spatial information can enhance the ability to represent the extracted features. There are two categories of methods to extract spectral–spatial information from HSI data. The first one extracts the spectral signatures and the spatial contexts separately, and then combines them to perform pixel-wise classification [9]. The second one treats the raw HSI data as a whole and extracts joint spatial–spectral features directly by using a 3D feature extractor. For example, spectral–spatial integrated features were extracted at different frequencies and scales using a series of 3D discrete wavelet filters [10], 3D Gabor wavelets [11], or 3D scattering wavelets [12]. Since hyperspectral data is typically presented in the format of 3D cubes, the second category of methods can result in a large number of discriminative features, which can effectively improve the classification performance.

In the above traditional approaches, handcrafted features are typically used, and they are expected to be discriminative and representative of the characteristics of HSI data. Typically, the extracted features are based on domain knowledge, which may lose some valuable details. In feature classification, Support Vector Machines (SVMs) [13] are often employed because SVMs are robust at representing high-dimensional vectors, but their capacity to represent is still limited to finite dimensions.

Since 2012, with the emergence of deep learning, the performance of many vision tasks has been dramatically improved, including but not limited to object detection [14], segmentation [15] and tracking [16]. In recent years, deep-learning-based methods have been introduced in the field of HSI classification. In particular, supervised convolutional neural networks (CNNs) and their extensions, including 1D-CNN [17,18], 2D-CNN [19,20], 3D-CNN [18,21], and ResNet [22,23], have been successfully employed to extract deep spectral–spatial features and have demonstrated state-of-the-art performance. Usually, a CNN consists of at least three convolutional layers for extracting both low-level and high-level features. Moreover, instead of separating feature extraction and feature classification as two steps, the CNN structure integrating feature extraction and feature classification into one framework through back-propagation [24]. Since the extracted features directly contribute to the final classification performance, deep learning methods achieve better performance than traditional methods.

However, two constraints limit the state-of-the-art deep CNNs from being used directly for HSI classification. The first factor is the different data format between RGB images and HSI. Specifically, the RGB images can be well represented by a 2D CNN model to extract features, while 3D CNN is preferable to preserve the abundant information being extracted from the spectral signatures and the spatial contexts of HSI. However, the number of parameters grows exponentially when the convolution moves from 2D to 3D [25]. A 3D CNN has a lot more parameters than a 2D counterpart due to its additional kernel dimension, making it more difficult and expensive to train. The second factor is the limited training sample dilemma. Generally, the feature representation ability of deep learning models strongly depends on a large number of training samples. However, the manual annotation for hyperspectral data is difficult, which results in the lack of labeled pixels. Without sufficient training samples, a deep model that has a powerful representation capacity may suffer from overfitting. Therefore, most of the existing CNN-based HSI classification methods focus on using small-scale models with relatively less depth (no more than 10 layers, generally) at the cost of a decrease in performance. However, leveraging large-scale networks is still desirable to jointly exploit underlying the nonlinear spectral and spatial structures of hyperspectral data residing in a high-dimensional feature space [26].

To address these inherent problems, in this paper, we propose a 3D asymmetric inception network (AINet) and a data-fusion transfer learning strategy for HSI classification, and our contributions can be summarized as four points:

1. A novel deep light-weight 3D CNN, AINet, with asymmetric structure is proposed to handle HSI classification, which uses the available small volume of HSI datasets to train the very deep neural network and fully exploit the potential of CNN.
2. Considering the properties of hyperspectral images as well as spectral signatures are emphasized over spatial contexts, an asymmetric inception unit (AI unit) is proposed. To convey and classify the features effectively, we replace the 3D convolution layer with two asymmetric inception units, namely the space inception unit and the spectrum inception unit.
3. Data fusion transfer learning is exploited to improve model initialization. It increases training efficiency and classification performance while compensating for data limitations.
4. The proposed method were tested on three public HSI datasets. The experimental results show that the proposed method achieves better performance than other state-of-the-art deep learning-based methods.

## 2. Related Works

### 2.1. Convolutional Neural Network Architectures

Convolutional neural network (CNN) is one of the most popular deep learning methods, and many CNN-based HSI classification methods have been presented in recent years. Three typical supervised CNN architectures, referred to as 1D, 2D, and 3D CNNs, were investigated in HSI classification. In 1D CNN-based HSI classification approaches, the kernels of a convolution layer convolve the input samples along the spectral dimension [17,18], and thus the spatial information is lost. The conventional way to obtain deep spectral-spatial representations by 2D CNNs is to train a model based on patch-based samples by expanding input data with more spatial information [27]. Meanwhile, HSI data are always compressed via a certain dimension-reduction algorithm, such as principal component analysis (PCA), and then convolved with 2D kernels. For instance, Makantasis et al. [28] exploited randomized PCA to condense the spectral dimensionality of the entire HSI first, followed by applying a 2D CNN to extract deep features from the compressed HSI. Furthermore, two stream CNN models are proposed to extract the spatial and spectral features separately. For instance, Zhang et al. [27] proposed a dual-channel CNN model where spectral features and spatial features are extracted via 1D CNN and 2D CNN respectively, and then a softmax regression classifier is used to combine these two kinds of features and predict classification results eventually. As the spatial features and spectral features are extracted separately, they may not fully exploit the joint spatial/spectral correlation information, which can be important for classification.

Since hyperspectral imagery is naturally a 3D data cube, it is reasonable to extract deep spectral-spatial features through 3D CNNs. The first 3D CNN network for HSI classification was proposed by Chen et al. [18] in 2016, and  $L_2$ -norm regularization and dropout are used. However, this is a shallow network, and it still suffers from overfitting when there is a shortage of annotated datasets. Similarly, a simpler 3D CNN structure using input cubes of HSIs with a smaller spatial size was presented in [21]. Later, Zhong et al. [22] proposed a supervised spectral-spatial residual network (SSRN) with consecutive spectral and spatial residual blocks to extract spectral and spatial features from HSI. Very recently, Fang et al. [29] proposed a 3D dense convolutional network with a spectral-wise attention mechanism (MSDN-SA) for HSI classification, where 3D dilated convolutions are exploited to capture spectral-spatial features at different scales, and all 3-D feature maps are densely connected to each other. The 3D CNN models generate classification maps with an approach that can directly process raw HSI. However, the classification accuracy decreases as the layers of the network become deeper. This is mainly due to the very limited HSI dataset used for training the network.

## 2.2. Efficient Deep Learning Models

Since AlexNet was proposed in 2012, a number of efficient deep learning models have been proposed. Three of these models, GoogLeNet [30], ResNet [23], and MobileNet [31], are related to our model proposed below. They also show the development trends of deep learning, with increasing depth while requiring less computation. GoogLeNet is the most basic of the so-called Inception series, ResNet is famous for its extreme depth, and MobileNet is well known for its low computation cost. The three models and their applications in HSI classification are described in detail below.

GoogLeNet consists of multiple inception modules, each of which contains four different convolution paths, and it is the most basic model of the Inception series [30]. Based on GoogLeNet, Inception-V1 to Inception-V4 are proposed [32–35]. The main advantage of an inception network is the ability to use multiple sizes of kernels for each branch, which allows the generation of a more flexible map of features [36]. Hidalgo et al. [37] proposed a data classification model that uses extended attribute profiles and an inception network to generate deep spatial–spectral features. Recently, a novel attention inception module was introduced to extract features dynamically from multiresolution convolutional filters [36].

ResNet employs shortcut connections to overcome the degradation problem, where accuracy becomes saturated and then degrades rapidly with the network depth increasing. In addition, in order to reduce the time complexity, He et al. [23] proposed a novel structure named “bottleneck”. Based on shortcut connection and the newly introduced bottleneck layers, He et al. [23] increased the depth of the network to more than 1000 layers and obtained excellent performance in image classification. Based on the shortcut connection, a supervised spectral–spatial residual network (SSRN) was proposed to mitigate the decreasing accuracy phenomenon and improve the HSI classification accuracy.

MobileNet employs depthwise separable convolutions to reduce the computation in the network and applies pointwise convolutions to combine the features of separate channels. Based on MobileNet-V1, MobileNet-V2 was also proposed to employ inverted residuals and linear bottlenecks, leading to better performance [31,38]. MobileNetV3 [39] is tuned through a combination of hardware-aware network architecture search (NAS) complemented by the NetAdapt algorithm and then enhanced by novel architecture advances. Some researchers have applied depthwise separable convolutions and pointwise convolutions to convolutional neural network architecture to improve HSI classification performance [40–42].

Our proposed asymmetric residual network not only benefits from the much deeper and light-weight network design, but also from the asymmetric inception unit that we tailored for the HSI dataset. Specifically, we propose an asymmetric inception unit (AI unit), which consists of the space inception unit and the spectrum inception unit, to convey and classify the features effectively.

## 2.3. Transfer Learning

Compared with the thousands of millions of annotated datasets used in vision tasks, annotated data in existing HSI datasets is insufficient. Moreover, the imbalance among HSI datasets of intraclass sets and those captured from different sensors also makes it challenging to train the neural network. In the computer vision community, one common solution to this problem is transfer learning. Transfer learning focuses on storing knowledge gained while solving one problem and applying it to a different but related problem [43]. It is defined as the ability of a system to recognize and apply knowledge and skills learned in previous tasks to a novel task [44]. The concept behind transfer learning is that, in deep neural networks, the bottom-level and middle-level features take up the majority of the parameters stored in the CNN model, and usually capture the textures and edges of the objects. Then, those low-level features designed for simple tasks such as detection can be reused for more complex tasks such as segmentation and tracking. A common strategy for transfer learning is to pretrain a model on one data set, where labeled samples are

sufficient, such as ImageNet, and then transfer the pretrained model to the target data set for fine-tuning.

Transfer learning offers two benefits: a better initialization of the model and a reduced training time for the network. It is beneficial to use transfer learning for data sets with very limited training samples, especially when the model is a deep CNN, which usually has a large number of parameters. Since the structure of HSI data is complex and the number of training samples is limited, transfer learning plays an instrumental role in HSI image classification. In [45], transfer learning has been adopted, but the source data sets and the target data sets are required to be gathered by the same sensor. Later, Lin et al. [46] used canonical correlation analysis (CCA) to transfer knowledge between two SAEs that were trained by source data and target data independently. Furthermore, the authors investigate the multisource or heterogeneous transfer learning strategy for HSI classification to alleviate the problem of small labeled samples [47,48]. In [49], Zhang et al. proposed a cross-modal transfer learning strategy which transfers models between data sets of different data modalities that exhibit different data characteristics, namely, from natural RGB image modality to HSI modality. It has been shown that the most significant benefit of the use of transfer learning is the improvement of model initialization, which is very important for training the model with limited samples.

Our proposed network adopts a data fusion transfer learning strategy. Concretely, the designed model is pretrained on HSI datasets captured by different sensors with 3D pyramid pooling and then fine-tuned on the target datasets to achieve a better performance.

### 3. Methodology

Among the deep learning models used in HSI literatures, 3D-CNN performs better than 2D-CNN for HSI classification due to the fact that 3D data formats are used in HSI. In fact, different objects in HSI generally have different spectral structures. Convolution along the spectral dimension is very critical. In addition, there are also some different objects which have similar spectral structures. For these objects, it is also beneficial to convolve along spatial dimensions to capture features, which can capture important spatial variations observed with high-resolution data [27,28]. For 2D-CNN based methods, without spectral dimension reduction, the number of parameters of 2D-CNNs will be extremely large due to the hundreds of bands. However, if dimension reduction is conducted, it may destroy the information of spectral structure which is critical for discriminating different objects.

Generally speaking, 3D-CNN-based approaches have better performance than 2D-CNN-based approaches [18,22]. However, the existing 3D-CNN-based approaches still have two deficiencies: (1) compared with 2D convolutions, 3D convolutions have more parameters and 3D-CNN models are computation-intensive; (2) being limited by the training samples in HSI datasets, 3D-CNN models employed in HSI classification almost always consist of less than five convolution layers. However, a large number of experiments in computer vision have proved that the deep depth of CNN is very significantly important for improving the performance of tasks related to image processing [23,30].

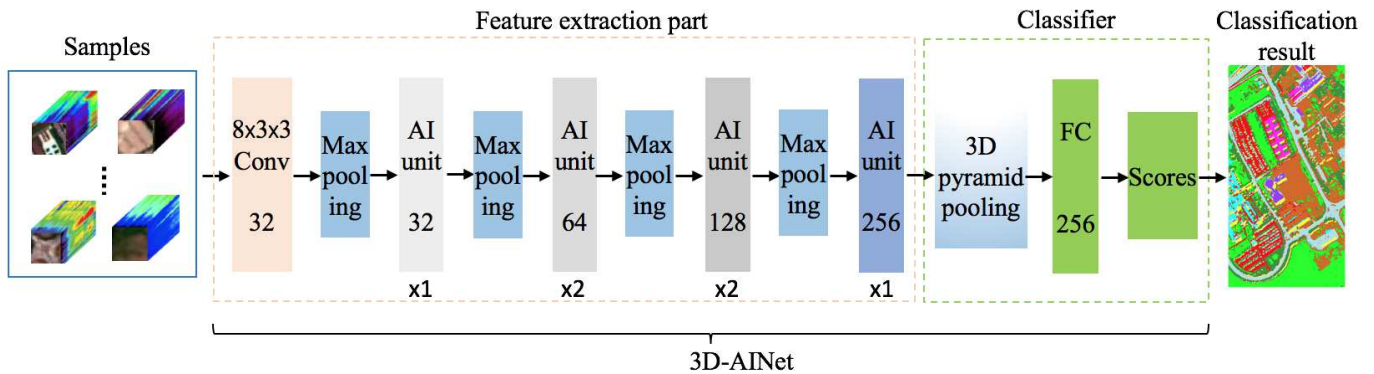
In this section, we first introduce the proposed AINet, and then describe the proposed data-fusion transfer learning strategy.

#### 3.1. AINet for HSI Classification

**Network Structure:** Figure 1 shows the overall framework of the proposed AINet for HSI classification. In order to utilize the spectral and spatial information contained in HSI, we extract  $L \times S \times S$ -sized cubes from raw HSI data as samples, where  $L$  and  $S$  indicate the number of spectrum bands and the spatial size accordingly (Following [18], we set  $S$  to 27 in this paper). Then, the samples are fed into AINet to extract deep spectral-spatial features, and finally the classification results are calculated. Inspired by the design of ResNet [23], AINet employs a similar basic structure and introduces some key modifications for tailoring on HSI dataset. AINet starts with a 3D convolution layer, then stacks six AI units of increasing widths. It connects one 3D spatial pyramid pooling and one fully



connected layer at the end. Specifically, the channels for the six AI units are 32, 64, 64, 128, 128 and 256, respectively. In order to reduce the dimension of features, four Max pooling layers are added with kernel = [3, 3, 3], stride = [2, 2, 2] within the six AI units.



**Figure 1.** Framework of AI-Net. On the left, the  $L \times S \times S$ -sized samples from the neighborhood window centered around each target pixel are extracted, and then the samples are fed into AI-Net to extract deep spectral–spatial features. Finally, the classification scores are calculated by the classifier.

**3D Pyramid Pooling:** Before the fully connected layer, a 3D pyramid pooling method is used to map features of different sizes to vectors with fixed dimensions. Different HSI datasets are usually captured by different sensors and with various numbers of spectrum bands, for example, the Pavia University dataset has 103 bands and the Indian Pines dataset contains 200 bands. With 3D pyramid pooling layer, the same network can be applied to different HSI datasets without any modification. In this paper, the 3D spatial pyramid pooling layer is composed of three-level pooling ( $1 \times 1 \times 1$ ,  $2 \times 1 \times 1$ ,  $3 \times 1 \times 1$ ). As the last AI unit has 256 channels, the outputs of 3D pyramid pooling layer are  $256 \times 6 \times 1 \times 1$ -sized cubes.

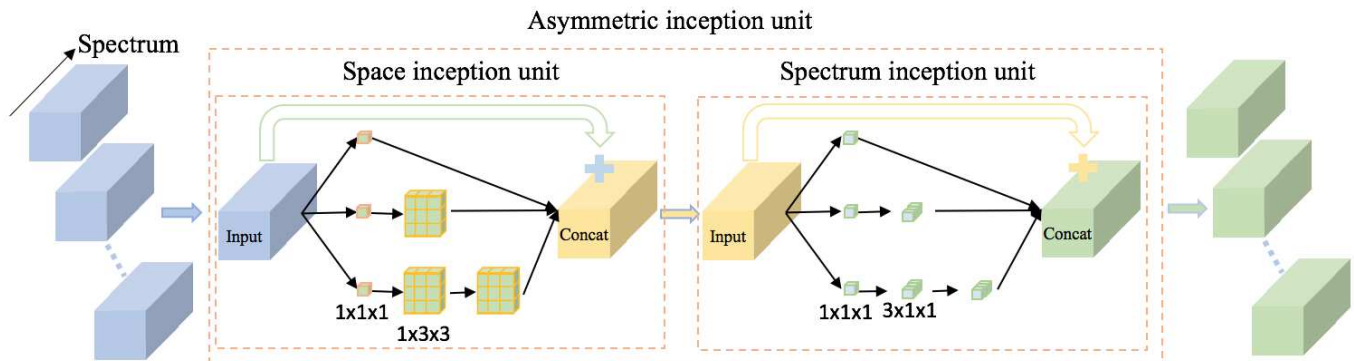
**Training and Loss:** We employ log softmax [50] as the activation function in the fully connected layer. During training, we take negative log likelihood as the loss function, and add  $L_2$  regularization term with weight  $1 \times 10^{-5}$  to the loss function for alleviating over-fitting. The optimizer is stochastic gradient descent (SGD) with momentum [51]. For all of the experiments, the same setting is adopted, where momentum, weight decay, batch size, epochs and learning rate are 0.9,  $1 \times 10^{-5}$ , 20, 60 and 0.01, respectively. In the last 12 epochs, the learning rate decreased to 0.001.

### 3.2. AI Unit

Because 3D convolution can learn the spectral and spatial information from the raw HSI datasets, the 3D-CNN based methods achieve the most advanced performance for HSI classification. However, compared with 2D convolutions, 3D convolutions are prone to overfitting and are computation-intensive. In order to address these problems, we propose an asymmetric inception unit (AI unit), which consists of the space inception unit and the spectrum inception unit. The structure of AI unit is illustrated in Figure 2.

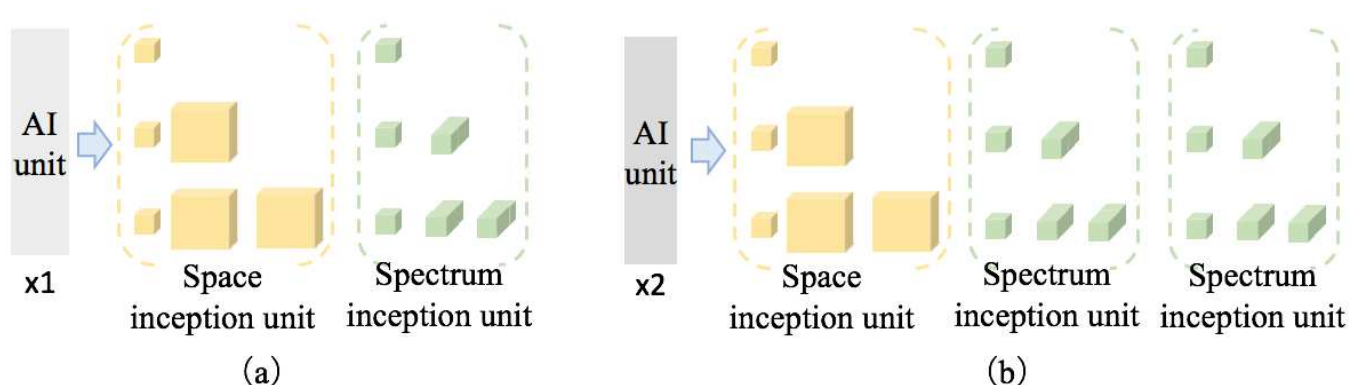
In the space inception unit, there are three space convolution paths. Path one has one pointwise convolution layer only, path two consists of one pointwise convolution layer and one 2D convolution layer with  $1 \times 3 \times 3$ -sized kernels, and path three has one pointwise convolution layer and two 2D convolution layers. The outputs of each path are concatenated in channel, and are added to the output of the shortcut connection. Inspired by the Inception networks [35], we set the three paths with different widths. For each unit, we set the widths of three paths with a split ratio 1:2:1. In the last two paths, the width of the pointwise convolution layer is half of that of the other convolution layers. For instance, in the AI unit with 32 channels, the width of the first path is 8. For the second path, the widths of the pointwise convolution layer and  $1 \times 3 \times 3$ -sized convolution layer are 8 and 16 respectively. The widths of the three layers of the last path are 4, 8 and 8 accordingly.

In the overall structure, the structure of spectrum inception unit is similar to the space inception unit, except that that  $1 \times 3 \times 3$ -sized 2D convolution layers in the space inception unit are replaced with  $3 \times 1 \times 1$ -sized 1D convolution layers.



**Figure 2.** Illustration of an AI unit. In AI unit, 3D convolution layer is replaced with two asymmetric inception units, i.e., space inception unit and spectrum inception unit. In the space inception unit, the input cube is fed into three different paths. In path one, a pointwise convolution layer is applied. In path two, one pointwise convolution layer and one 2D convolution layer are used. In path three, one pointwise convolution layer and two 2D convolution layers are used. The outputs of each path are concatenated in channel, and are added to the output of the shortcut connection. The structure of spectrum inception unit is similar to the space inception unit, except that  $1 \times 3 \times 3$ -sized convolution layers are replaced with  $3 \times 1 \times 1$ -sized convolution layers in spectrum inception unit.

In HSI datasets, the spectral resolution is much higher than the spatial resolution, and the spectral information is much richer. Therefore, in the process of spectral–spatial features extraction, we pay more attention to spectral feature extraction. In the proposed AINet, there are six AI units. The four units located in the middle can be divided into two groups, and each group stacks two units of equal width. Here, instead of stacking two same AI units in each group, we stack one space inception unit and two spectrum inception units. This is different from some popular networks, such as ResNet [23] and MobileNet [31], which build the whole model by stacking the same units. Figure 3 shows the difference between one AI unit and two AI units.



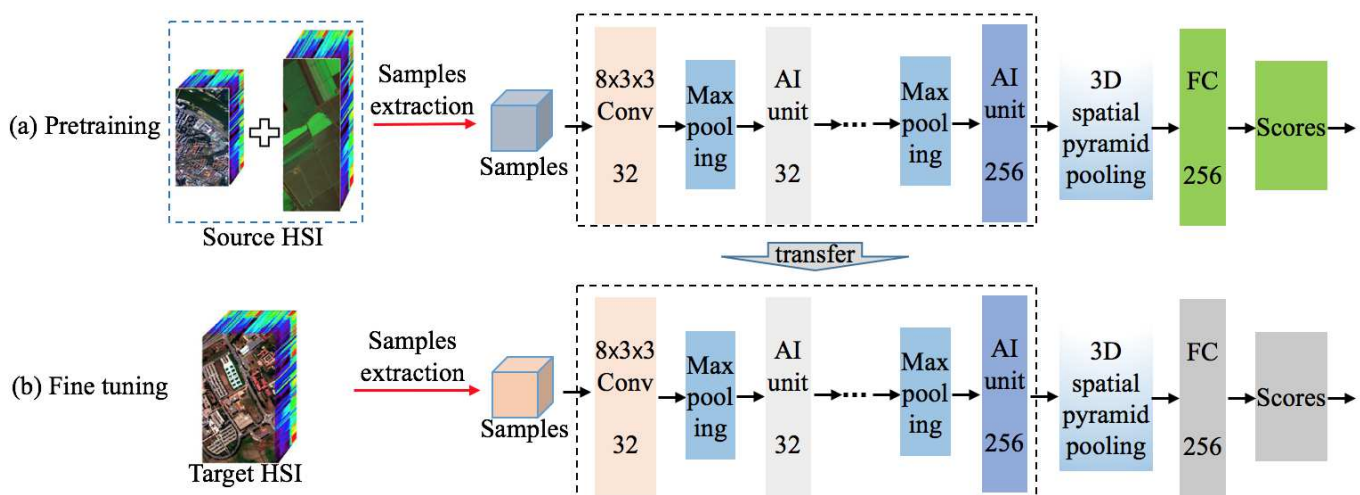
**Figure 3.** Illustration of stacking two AI units. (a) AI unit  $\times 1$ ; (b) AI unit  $\times 2$ . Instead of stacking two AI units with the same type, we stack one space inception unit and two spectrum inception units to form AI unit  $\times 2$  as shown in (b).

### 3.3. Transfer Learning with Data Fusion

In RGB images classification, pretraining networks on the ImageNet dataset which has over 14 million hand-annotated images and over 20,000 categories is common, and it is very useful for improving the performance and overcoming the problem of limited

training samples. The diversity of datasets used for pretraining is a key factor in transfer learning. For example, pretraining the same model on a dataset with a million images and a thousand categories always achieves better results than pretraining the same model on a dataset with 10 million images and 10 categories. We believe that model pretraining with more diverse samples may result in better generalization ability.

For further improving the performance of HSI classification, we propose a data-fusion transfer learning strategy. As shown in Figure 4, the strategy is composed of data-fusion pretraining and finetuning: (1) data-fusion pretraining—during pretraining, the proposed network is trained on two different HSI datasets to improve the diversity of samples and obtain a robust initialized model; (2) fine-tuning—after the pretrained model is acquired, the new model is initialized using the parameters of the pretrained model for the target HSI dataset. The fully connected layers of the proposed model are randomly initialized with a Gaussian distribution.



**Figure 4.** Data-fusion-based transfer learning. (a) Data-fusion pretraining: during pretraining, the proposed network is trained on two different HSI datasets for improving the diversity of samples and obtaining a robust initialized model. (b) Fine-tuning: after the pretrained model is acquired, the new model is initialized using the parameters of the pretrained model for the target HSI dataset. Here, the fully connected layers of the proposed model are randomly initialized with a Gaussian distribution.

During pretraining, the proposed network is trained on two source HSI datasets. Here, Pavia Center dataset and Salinas dataset are used as source HSI datasets for pretraining. Among the several public HSI datasets, those two datasets have the largest number of labeled samples. To be more specific, the model is initialized with Gaussian distribution on one-source HSI dataset and pretrained for  $N$  epochs, and then the feature extraction part is fixed and the classifier is reinitialized with Gaussian distribution. Later on, the feature extraction part and classifier on the other source HSI dataset are pretrained for  $\frac{N}{2}$  epochs with a different learning rate. In this paper,  $N$  is set to 10 and the learning rate used for the feature extraction part is tenth of that used for the second pretraining HSI dataset.

After pretraining the model on the two source HSI datasets, we transfer the entire model except for the classifier, to construct the fine-tuning model for initialization of the target HSI dataset. Then the transfer part and the new classifier are fine-tuned at the same learning rate for training the second source HSI dataset.

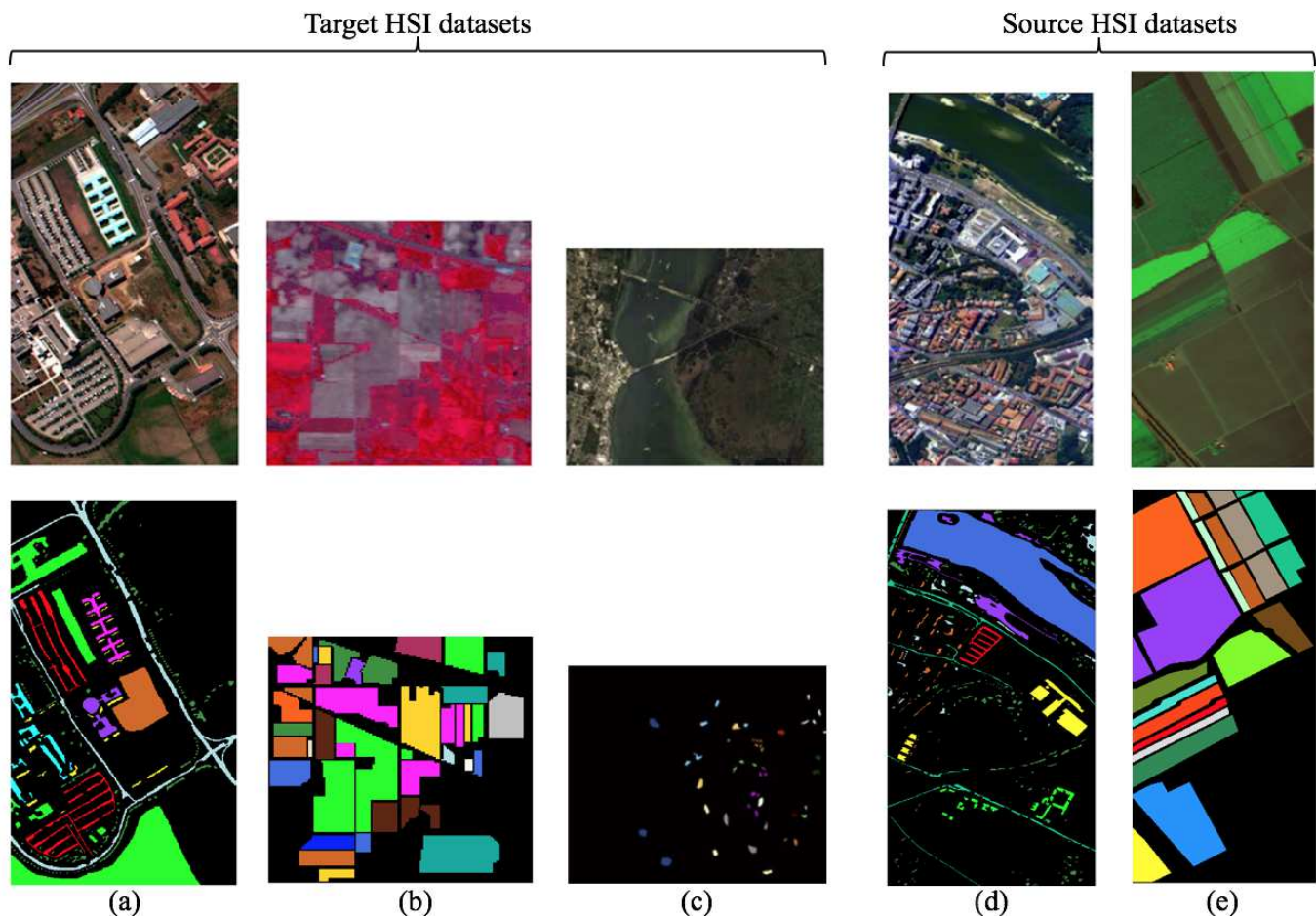
## 4. Experiments

### 4.1. Datasets and Experiments Setting

In this paper, we compare the proposed AINet with a traditional approach and five CNN-based approaches for HSI classification on three public HSI datasets, including Pavia University, Indian Pines and KSC. In the transfer-learning experiment, the Pavia Center dataset and the Salinas dataset are employed as the source datasets. The false-color



composite and ground truth of each dataset are shown in Figure 5. A brief introduction of each dataset is given in the following part and more information can be found on the website [http://www.ehu.es/ccwintco/index.php/Hyperspectral\\_Remote\\_Sensing\\_Scenes](http://www.ehu.es/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes) (accessed on 20 February 2022). The code of the proposed algorithm can be found at: <https://github.com/UniLauX/AINet> (accessed on 20 February 2022).



**Figure 5.** False-color composites (first row) and ground truths (second row) of experimental HSI datasets. Each color represents one kind of object. (a) Pavia University; (b) Indian Pines; (c) Kennedy Space Center; (d) Pavia Center; (e) Salinas.

Pavia University and Pavia Center datasets were captured by Reflective Optics System Imaging Spectrometer (ROSIS) sensor in 2001. After several noisiest bands being removed, Pavia University has 103 bands and Pavia Center has 102 bands. Both datasets are divided into 9 classes.

Indian Pines and Salinas datasets were acquired by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor in 1992. After correction, each dataset has 200 bands and contains 16 classes.

KSC was acquired by the AVIRIS sensor in 1996, and after removing water absorption and low SNR bands, 176 bands were used for analysis. For classification purposes, 13 classes are defined.

For the three target HSI datasets, samples are divided into training samples and testing samples. For comparison purposes, we follow [18] to set the samples distribution for Indian Pines and KSC datasets. As for the Pavia University dataset, 200 random samples are taken from each class as training samples. Tables 1–3 provide the split details. For the two-source HSI datasets, the description of the two datasets is shown in Tables 4 and 5.

**Table 1.** Samples distribution for Pavia University dataset.

No.	Class Name	Training Samples	Test Samples
1	Asphalt	200	6431
2	Meadows	200	18,449
3	Gravel	200	1899
4	Trees	200	2899
5	Painted metal sheets	200	1145
6	Bare Soil	200	4829
7	Bitumen	200	1130
8	Self-Blocking Bricks	200	3482
9	Shadows	200	747
total		1800	40976

**Table 2.** Samples distribution for Indian Pines dataset.

No.	Class Name	Training Samples	Test Samples
1	Alfalfa	30	16
2	Corn—notill	150	1198
3	Corn—mintill	150	232
4	Corn	100	5
5	Grass—pasture	150	139
6	Grass—trees	150	580
7	Grass—pasture-mowed	20	8
8	Hay—windrowed	150	130
9	Oats	15	5
10	Soybean—notill	150	675
11	Soybean—mintill	150	2032
12	Soybean—clean	150	263
13	Wheat	150	55
14	Woods	150	793
15	Buildings—Grass—Trees—Drives	50	49
16	Stone—Steel—Towers	50	43
total		1765	6223

**Table 3.** Samples distribution for KSC dataset.

No.	Class Name	Training Samples	Test Samples
1	Scrub	33	314
2	Willow Swamp	23	220
3	Cabbage Palm Hammock	24	232
4	Cabbage Palm / Oak Hammock	24	228
5	Slash Pine	15	146
6	Oak / Broadleaf Hammock	22	207
7	Hardwood Swamp	9	96
8	Graminoid Marsh	38	352
9	Spartina Marsh	51	469
10	Cattail Marsh	39	365
11	Salt Marsh	41	378
12	Mud Flats	49	454
13	Water	91	836
total		459	1297

In the transfer learning experiment, we randomly extracted 200 samples from each class of Pavia Center dataset, 100 samples from each category of Salinas dataset as test samples, and take the rest as training samples.

**Table 4.** Samples distribution for Pavia Center dataset.

No.	Class Name	Samples
1	Water	824
2	Trees	820
3	Asphalt	816
4	Self-Blocking Bricks	808
5	Bitumen	808
6	Tiles	1260
7	Shadows	476
8	Meadows	824
9	Bare Soil	820
total		7456

**Table 5.** Samples distribution for Salinas dataset.

No.	Class Name	Samples
1	Brocoli_green_weeds_1	2009
2	Brocoli_green_weeds_2	3726
3	Fallow	1976
4	Fallow_rough_plow	1394
5	Fallow_smooth	2678
6	Stubble	3959
7	Celery	3579
8	Grapes_untrained	11,271
9	Soil_vinyard_develop	6203
10	Corn_senesced_green_weeds	3278
11	Lettuce_romaine_4wk	1068
12	Lettuce_romaine_5wk	1927
13	Lettuce_romaine_6wk	916
14	Lettuce_romaine_7wk	1070
15	Vinyard_untrained	7268
16	Vinyard_vertical_trellis	1807
total		54,129

#### 4.2. Performance Comparison of Different Network Structures

In this section, we compare the proposed AINet with a traditional method and five CNN-based HSI classification methods, that are SVM-3DG [52], 1D-CNN, 2D-CNN, 3D-CNN [18], MSDN-SA [29], SSRN [22]. The experiments with the same settings are ran for 5 times to obtain the average performance. The experimental results are listed in Tables 6–8, where the number of training samples, the number of parameters used in the convolution layers, the depth of CNN models, overall accuracy (OA), average accuracy (AA) and kappa coefficient ( $K$ ) are reported. OA is the ratio between the number of correctly classified samples in the test set and the total number of test sets. AA is the mean of the OA of all the categories.  $K$  is a coefficient which measures inter-rater agreement for qualitative items [53]. The classification maps are shown in Figures 6–8. From Tables 6–8, we can see that the proposed AINet achieves the highest classification performance on all of the datasets. For instance, in the Indian Pines dataset, OA of AINet is 99.14, which is 9.15% better than that of 2D-CNN, 1.58% better than that of 3D-CNN and 0.74 better than that of SSRN. The experiments indicate that all of the 3D-CNN-based HSI classification methods are superior to 2D-CNN. From 3D-CNN, MSDN-SA, SSRN to AINet, the depth of the

models is increasing and the classification accuracy keeps improving. In particular, the depths of the four models are 4, 7, 12, 32 respectively. Although AINet is much deeper than SSRN, AINet has slightly more parameters than SSRN and much fewer than 3D-CNN.

**Table 6.** Classification results for the Pavia University dataset.

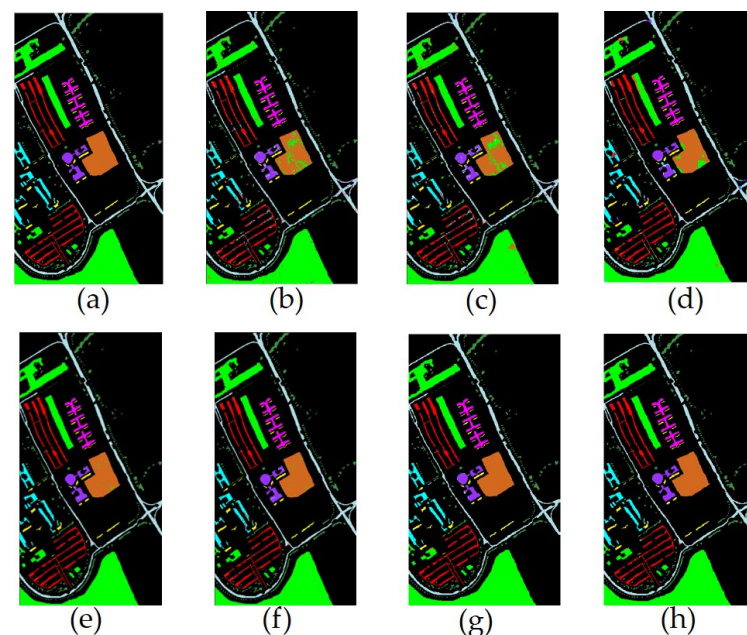
Models	SVM-3DG [52]	1D-CNN [18]	2D-CNN [18]	3D-CNN [18]	MSDN-SA [29]	SSRN [22]	AINet
# train	3930	3930	3930	3930	3930	<b>1800</b>	<b>1800</b>
# param.	\	<b>2898</b>	0.183 M	5.849 M	3.058 M	0.453 M	0.487 M
depth	\	4	4	4	7	12	32
OA	90.18 ± 0.95	89.01 ± 1.31	91.13 ± 1.49	95.63 ± 0.79	96.85 ± 0.71	98.98 ± 0.73	<b>99.42 ± 0.89</b>
AA	91.47 ± 0.90	89.15 ± 0.87	92.58 ± 1.77	95.67 ± 0.86	97.36 ± 0.38	99.07 ± 1.46	<b>99.51 ± 0.58</b>
K	87.39 ± 0.96	87.47 ± 1.60	89.63 ± 0.94	95.38 ± 1.58	95.85 ± 0.93	98.64 ± 1.31	<b>99.22 ± 1.73</b>

**Table 7.** Classification results for the Indian Pines dataset.

Models	SVM-3DG [52]	1D-CNN [18]	2D-CNN [18]	3D-CNN [18]	MSDN-SA [29]	SSRN [22]	AINet
# train	1765	1765	1765	1765	1765	1765	1765
# param.	\	<b>25,920</b>	0.183 M	44.893 M	3.058	0.453 M	0.487 M
depth	\	6	4	4	7	12	32
OA	85.87 ± 0.91	87.81 ± 1.28	89.99 ± 1.62	97.56 ± 1.21	98.02 ± 1.85	98.40 ± 0.90	<b>99.14 ± 1.74</b>
AA	89.74 ± 0.82	93.12 ± 0.86	97.19 ± 1.96	99.23 ± 1.94	98.69 ± 0.94	98.52 ± 1.98	<b>99.47 ± 1.64</b>
K	84.08 ± 1.54	85.30 ± 1.69	87.95 ± 0.86	97.02 ± 1.97	97.75 ± 1.21	98.14 ± 0.75	<b>99.00 ± 1.27</b>

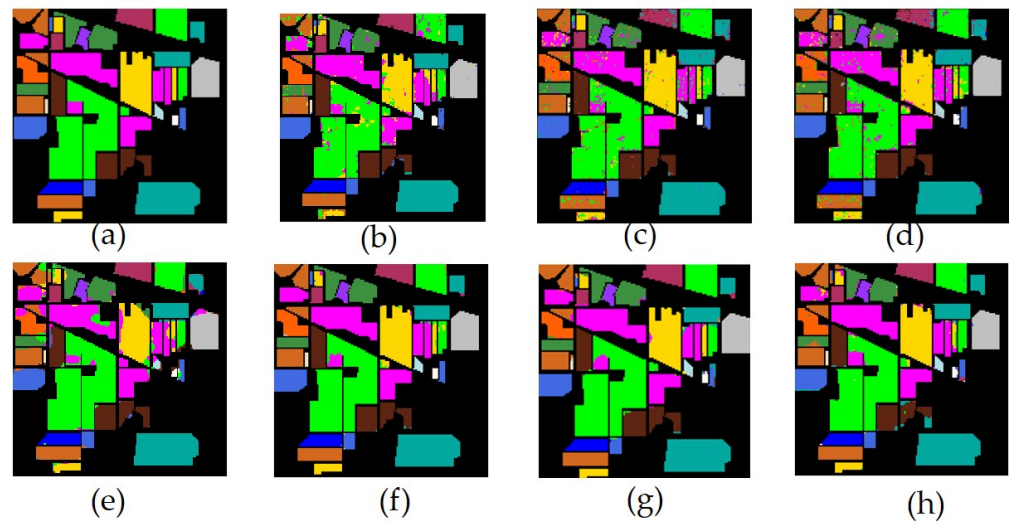
**Table 8.** Classification results for the KSC dataset.

Models	SVM-3DG [52]	1D-CNN [18]	2D-CNN [18]	3D-CNN [18]	MSDN-SA [29]	SSRN [22]	AINet
# train	459	459	459	459	459	459	459
# param.	\	<b>14,904</b>	0.183 M	5.849 M	3.058 M	0.453 M	0.487 M
depth	\	5	4	4	7	12	32
OA	88.24 ± 1.36	89.23 ± 1.69	94.11 ± 1.36	96.31 ± 0.98	97.95 ± 1.91	98.65 ± 1.37	<b>99.01 ± 0.69</b>
AA	85.68 ± 1.87	83.32 ± 1.05	91.98 ± 1.19	94.68 ± 2.04	97.80 ± 1.94	97.78 ± 1.32	<b>98.65 ± 0.59</b>
K	87.04 ± 0.65	86.91 ± 1.47	93.44 ± 0.98	95.90 ± 1.08	97.70 ± 1.65	98.54 ± 0.89	<b>98.90 ± 1.15</b>

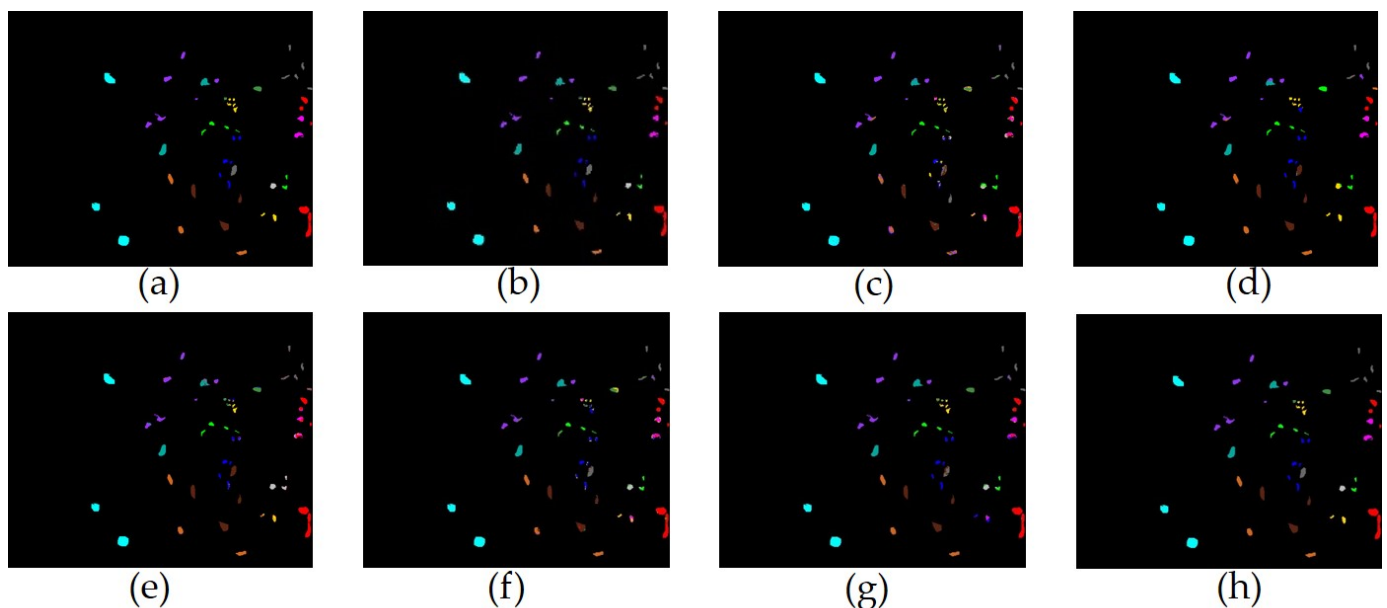


**Figure 6.** Classification maps for Pavia University dataset. (a) Ground-truth map; (b) SVM-3DG; (c) 1D-CNN; (d) 2D-CNN; (e) 3D-CNN; (f) MSDN-SA; (g) SSRN; (h) AINet.





**Figure 7.** Classification maps for Indian Pines dataset. (a) Ground-truth map; (b) SVM-3DG; (c) 1D-CNN; (d) 2D-CNN; (e) 3D-CNN; (f) MSDN-SA; (g) SSRN; (h) AINet.



**Figure 8.** Classification maps for KSC dataset. (a) Ground-truth map; (b) SVM-3DG; (c) 1D-CNN; (d) 2D-CNN; (e) 3D-CNN; (f) MSDN-SA; (g) SSRN; (h) AINet.

#### 4.3. Classification Results with Spatially Disjoint Samples

Previous research [4,54,55] has pointed out that the random-sampling strategy has a significant impact on the reliability and quality of the solution, since this may make it easier for the networks to classify the test samples during the inference stage (as the network has already processed them in some way during training). As compared to disjointed samples, randomly selected samples may result in significant spatial overlap of the training and test samples, which may overestimate classification performance. Because of this, the results obtained by the model may not be realistic, since artificially optimistic results may be obtained. To obtain more realistic results and a more accurate evaluation of the models, in this subsection, a sampling strategy based on selecting spatially separated samples is used to evaluate the model. The classification results on two sampling strategies of all compared methods in Section 4.2 are summarized in Table 9–11.

**Table 9.** Classification results with spatially disjoint samples for the Pavia University dataset.

Models	SVM-3DG [52]	1D-CNN [18]	2D-CNN [18]	3D-CNN [18]	MSDN-SA [29]	SSRN [22]	AINet
OA	79.97 ± 1.96	79.18 ± 1.39	75.41 ± 1.79	77.25 ± 1.08	76.88 ± 2.14	81.98 ± 1.24	<b>83.04 ± 1.28</b>
AA	80.99 ± 0.79	79.47 ± 0.86	76.56 ± 1.36	78.56 ± 2.14	78.07 ± 1.35	83.67 ± 2.43	<b>85.64 ± 0.86</b>
K	78.53 ± 1.42	77.31 ± 1.38	74.14 ± 1.03	75.61 ± 1.62	75.78 ± 1.94	80.49 ± 0.87	<b>82.49 ± 1.36</b>

**Table 10.** Classification results with spatially disjoint samples for the Indian Pines dataset.

Models	SVM-3DG [52]	1D-CNN [18]	2D-CNN [18]	3D-CNN [18]	MSDN-SA [29]	SSRN [22]	AINet
OA	77.54 ± 1.36	78.21 ± 1.79	76.06 ± 1.54	75.49 ± 2.16	78.29 ± 0.46	79.94 ± 1.84	<b>85.97 ± 1.08</b>
AA	79.92 ± 1.24	79.72 ± 1.56	78.70 ± 0.89	76.66 ± 1.18	79.21 ± 0.79	80.78 ± 2.06	<b>87.09 ± 1.27</b>
K	76.91 ± 0.98	76.32 ± 0.93	74.26 ± 1.28	73.87 ± 1.78	75.78 ± 1.47	76.47 ± 1.19	<b>83.14 ± 0.86</b>

**Table 11.** Classification results with spatially disjoint samples for the KSC dataset.

Models	SVM-3DG [52]	1D-CNN [18]	2D-CNN [18]	3D-CNN [18]	MSDN-SA [29]	SSRN [22]	AINet
OA	80.64 ± 1.58	79.36 ± 1.50	77.54 ± 1.26	79.10 ± 1.23	79.03 ± 1.36	78.12 ± 1.07	<b>80.92 ± 1.02</b>
AA	82.74 ± 1.24	80.94 ± 1.22	79.65 ± 1.24	82.28 ± 0.98	82.93 ± 1.20	80.77 ± 1.65	<b>83.26 ± 1.27</b>
K	78.48 ± 0.68	77.95 ± 1.81	76.34 ± 2.04	<b>78.77 ± 1.42</b>	77.41 ± 0.96	75.78 ± 1.34	77.57 ± 1.81

As can be seen, 2D-CNN, 3D-CNN, MSDN-SA and SSRN suffer an accuracy deterioration. In addition, the performance of 2D-CNN and 3D-CNN endures a drastic decline. As the spatial resolution of the Indian dataset is lower than that of the other two datasets, 2D-CNN and 3D-CNN algorithms that focus more on spatial information decline significantly in this dataset. Although AINet also experiences performance degradation, it still achieves the highest OA, AA and K.

#### 4.4. Results of Transfer Learning

In this section, we combine the proposed AINet with data-fusion-based transfer learning to further improve the classification performance. In [45], the authors adopted transfer learning in their framework, but restricts that the data used for pretraining must be collected by the same sensor as the target data. In contrast to previous work, we have not imposed restrictions on the datasets used for pretraining, which makes these results more applicable than previous works.

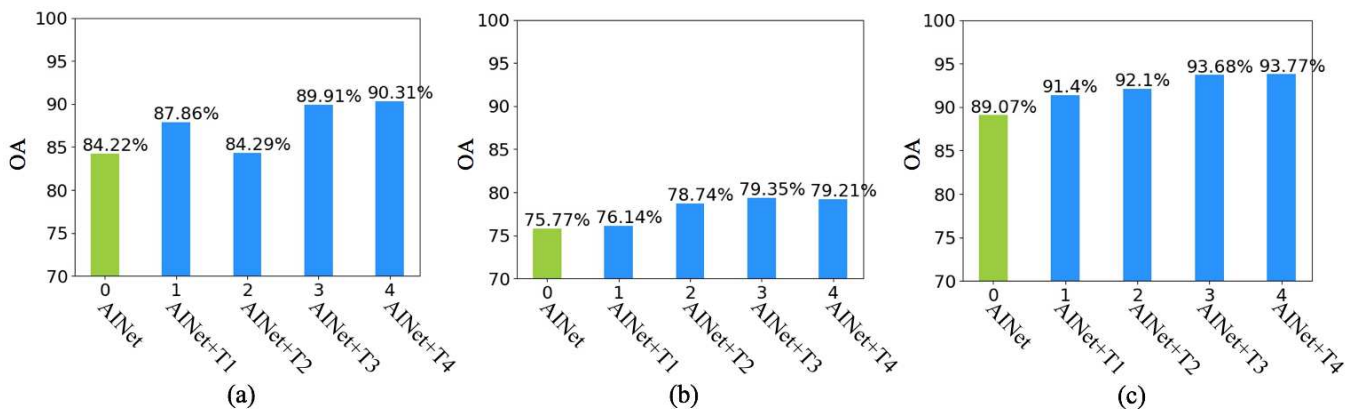
Here, we employ five HSI datasets in total. Three datasets, Pavia University, Indian Pines and KSC, are used as target datasets. Two datasets, Pavia Center and Salinas, are used as source datasets. Both the source dataset Pavia Center and the target dataset Pavia University were collected by the same sensor ROSIS, so their spatial and spectral properties are similar. The source dataset Salinas and the target dataset Indian Pines were taken by the same sensor AVIRIS and their spatial and spectral resolution are roughly identical. The last target dataset, KSC, was also collected by AVIRIS, but KSC has 176 bands, which is much more than Salinas and Indian Pines. As a result, the basic attributes involved in KSC are rather different from those in Salinas and Indian Pines.

In transfer-learning experiments, we implement the experiments with four different transfer-learning strategies, named AINet+T1, AINet+T2, AINet+T3 and AINet+T4, respectively. In AINet+T1, we pretrain the proposed model with Pavia Center data at first, then transfer the pretrained model to target datasets and fine-tune it on target datasets. Similarly, in AINet+T2, we firstly pretrain our proposed model on Salinas, then transfer and fine-tune the pretrained model to target datasets. Different from AINet+T1 and AINet+T2, both AINet+T3 and AINet+T4 have two pretraining stages, in which different source datasets are used for pretraining. In AINet+T3, we pretrain the model on Pavia Center dataset in the first stage and pretrain the model on Salinas dataset in the second stage. In AINet+T4, we inverse the order of using source datasets to pretrain.

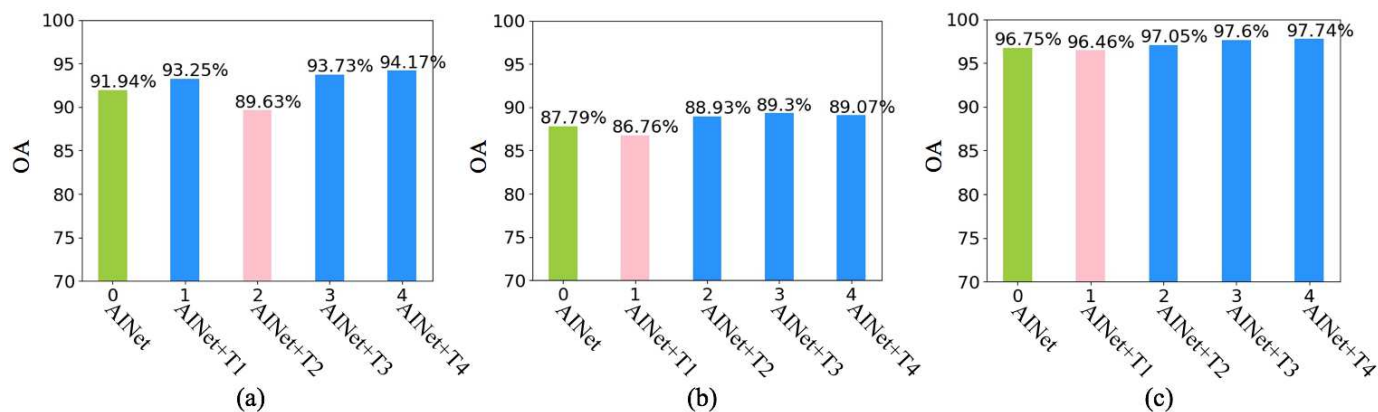
The experimental results of transfer learning are listed in Table 12 and shown in Figures 9 and 10. For each target dataset, we randomly choose 15 and 30 samples from each class as the training samples and reserve the rest as test samples.

**Table 12.** Transfer learning results for the three target datasets.

Training Samples	15			30		
<b>Dataset</b>	<b>Pavia University</b>					
	OA	AA	K	OA	AA	K
AINet	84.22 ± 0.72	86.36 ± 0.89	79.67 ± 1.34	91.94 ± 0.87	93.52 ± 0.90	89.48 ± 0.82
AINet+T1	87.86 ± 0.69	88.94 ± 0.84	84.20 ± 0.76	93.25 ± 0.77	95.06 ± 0.98	91.17 ± 0.85
AINet+T2	84.29 ± 0.58	84.02 ± 1.19	79.64 ± 0.29	89.63 ± 0.59	90.41 ± 1.30	86.22 ± 0.79
AINet+T3	89.91 ± 1.67	89.58 ± 1.49	86.80 ± 0.89	93.73 ± 0.46	94.04 ± 1.07	91.77 ± 1.20
AINet+T4	<b>90.31 ± 1.23</b>	<b>90.57 ± 0.85</b>	<b>87.32 ± 0.78</b>	<b>94.17 ± 1.16</b>	<b>94.52 ± 1.39</b>	<b>92.32 ± 0.89</b>
<b>Dataset</b>	<b>Indian Pines</b>					
	OA	AA	K	OA	AA	K
AINet	75.77 ± 1.82	86.43 ± 0.95	72.68 ± 1.79	87.79 ± 1.29	93.76 ± 0.98	86.12 ± 1.84
AINet+T1	76.14 ± 1.69	86.37 ± 1.54	73.14 ± 1.83	86.76 ± 1.65	93.21 ± 0.87	84.92 ± 0.75
AINet+T2	78.74 ± 0.74	88.09 ± 0.99	76.10 ± 1.18	88.93 ± 1.89	94.32 ± 1.05	87.42 ± 1.48
AINet+T3	<b>79.35 ± 1.37</b>	88.00 ± 1.79	<b>76.70 ± 1.74</b>	<b>89.30 ± 0.87</b>	<b>94.34 ± 0.85</b>	<b>87.83 ± 1.24</b>
AINet+T4	79.21 ± 0.47	<b>88.39 ± 0.49</b>	76.55 ± 1.13	89.07 ± 0.28	94.29 ± 0.86	87.64 ± 1.76
<b>Dataset</b>	<b>KSC</b>					
	OA	AA	K	OA	AA	K
AINet	89.07 ± 0.67	88.61 ± 0.31	87.83 ± 0.87	96.75 ± 1.98	96.14 ± 0.74	96.36 ± 0.66
AINet+T1	91.40 ± 1.65	89.80 ± 1.49	90.39 ± 0.59	96.46 ± 1.28	96.07 ± 1.49	96.13 ± 0.89
AINet+T2	92.10 ± 1.62	91.75 ± 2.17	91.21 ± 0.91	97.05 ± 0.52	97.48 ± 0.58	96.70 ± 1.38
AINet+T3	93.68 ± 2.07	<b>93.48 ± 1.49</b>	92.97 ± 0.65	97.60 ± 1.20	97.71 ± 0.48	97.31 ± 1.27
AINet+T4	<b>93.77 ± 0.55</b>	93.03 ± 0.91	<b>93.06 ± 0.52</b>	<b>97.74 ± 1.57</b>	<b>97.93 ± 2.21</b>	<b>97.87 ± 1.26</b>



**Figure 9.** Transfer learning experiments with 15 training samples per class. (a) Pavia University; (b) Indian Pines; (c) Kennedy Space Center.



**Figure 10.** Transfer learning experiments with 30 training samples per class. (a) Pavia University; (b) Indian Pines; (c) Kennedy Space Center.

## 5. Discussion

### 5.1. Assessment of the Asymmetric Inception Unit

In order to evaluate the performance of the asymmetric inception unit (AI Unit) in the proposed framework, we replace the AI unit in the AINet with the residual unit as the basic model. The details of the AI unit have been introduced previously (Section 3.2). In addition, as described in Section 3.2, instead of stacking two AI units with the same type, we stack one space inception unit and two spectrum inception units to form AI unit  $\times 2$ . To verify the performances of the basic network model, AINet (AI unit  $\times 1$ ) and AINet (AI unit  $\times 2$ ), we apply them to three target data sets. For the three datasets, the number of training samples is the same as Section 4.2. Table 13 list the experimental results. From Table 13 we can clearly see that AI Unit improves the classification results for all three datasets. Performance is boosted by a larger margin on the Indian Pines and KSC datasets than on the Pavia University dataset. These tables jointly demonstrate the effectiveness of AINet, being capable of providing the highest performance regarding a range of criteria, including OA, AA and K. From the basic model to AINet (AI unit  $\times 2$ ), the performance increases step by step. We argue that this is because the structure of AI Unit employed is becoming more effective.

**Table 13.** Classification results for the three target dataset.

Dataset		Pavia University		
Training Samples		1800		
Models	Basic network model	AINet (AI unit $\times 1$ )	AINet (AI unit $\times 2$ )	
OA	$99.27 \pm 1.24$	$99.36 \pm 0.54$	$99.42 \pm 0.89$	
AA	$99.39 \pm 1.41$	$99.44 \pm 0.68$	$99.51 \pm 0.58$	
K	$99.08 \pm 1.60$	$99.11 \pm 0.46$	$99.22 \pm 1.73$	
Dataset		Indian Pines		
Training Samples		1765		
Models	Basic network model	AINet (AI unit $\times 1$ )	AINet (AI unit $\times 2$ )	
OA	$98.85 \pm 2.13$	$99.00 \pm 0.90$	$99.14 \pm 0.74$	
AA	$99.52 \pm 1.24$	$99.30 \pm 0.31$	$99.47 \pm 1.64$	
K	$98.67 \pm 1.13$	$98.71 \pm 0.82$	$99.00 \pm 1.27$	
Dataset		KSC		
Training Samples		459		
Models	Basic network model	AINet (AI unit $\times 1$ )	AINet (AI unit $\times 2$ )	
OA	$97.12 \pm 0.38$	$98.29 \pm 1.04$	$99.01 \pm 0.69$	
AA	$96.47 \pm 0.88$	$97.01 \pm 0.87$	$98.65 \pm 0.59$	
K	$97.01 \pm 0.94$	$97.15 \pm 1.27$	$98.90 \pm 1.15$	



### 5.2. Assessment of the Data Fusion Transfer Learning

The experimental results of transfer learning are listed in Table 12 and shown in Figures 9 and 10. From the experimental results, we can see that transfer-learning strategies are beneficial for improving the performance of AINet, especially when the available training samples are relatively small. When we extract 15 samples per class for training, the transfer-learning strategy AINet+T1 obtains OA gains of 3.64% for Pavia University, 0.37% for Indian Pines and 2.33% for KSC, respectively. The gain provided by transfer learning drops with the increase in training samples. We conjecture that as the number of training samples increases, the model can directly obtain more guidance information from the target HSI data set. Therefore, the AINet can work well even without transfer learning.

Compared with pretraining the model with a single source dataset, pretraining the model with multiple source datasets is more effective. As we can see from Table 12, excellent performances are always achieved by AINet+T3 and AINet+T4, which fuse two different source datasets in the pretraining stage. For instance, in Pavia University, AINet+T4 improved the OA from 84.22% to 90.31% (improved by 6.09%). However, AINet+T1 just improved the OA to 87.86%, 2.45 percentage points lower than that of AINet+T4. We conjecture that this is mainly because the model pretrained with a multiple-source dataset has a better generalization ability than the model pretrained with a single-source dataset. From Figure 10, we can see that when we increase the number of training samples to 30 per class, pretraining the model with a single heterogeneous dataset (the dataset collected by different sensors) may harm the performance, but, pretraining the model with multiple-source datasets still boosts the performance.

## 6. Conclusions

This paper proposes a 3D asymmetric inception network (AINet) for hyperspectral image classification. Firstly, compared to traditional 3D CNNs, AINet proposed a light-weight but much deeper architecture that can exploit the potential of deep learning to extract representative features while alleviating the problems caused by limited annotated datasets. Secondly, considering the property of hyperspectral images, spectral signatures are emphasized over spatial contexts in the proposed AI Unit. Furthermore, a data-fusion transfer learning strategy is adopted to improve the initialization of the model and the classification accuracy.

We conduct comparison experiments on three challenging public HSI datasets and compare our proposed AINet with deep learning based HSI classification methods. The results of comparison experiments have demonstrated that our proposed AINet achieves competitive performance with others. Although AINet is much deeper than SSRN, the parameters of AINet are slightly more than that of SSRN and much less than that of 3D-CNN. In fact, benefiting from the AI Unit, AINet contains much less parameters and higher performance than the basic network model. In addition, we have performed experiments to verify the effectiveness of our proposed data fusion transfer learning strategy. Results show that compared with pretraining the model with a single-source dataset, pretraining the model with multiple-source datasets is more effective.

In the future, there are two topics we are keen to pursue. Investigating the reduction of the training time brought by transfer learning is the first, and the second is taking use of some policies to overcome the data imbalance in HSI classification.

**Author Contributions:** Conceptualization, B.F., Y.L. and H.Z.; data curation, B.F. and Y.L.; investigation, B.F. and Y.L.; methodology, B.F. and H.Z.; validation, B.F. and Y.L.; visualization, B.F.; writing—original draft, H.Z.; writing—review and editing, B.F. and Y.L.; supervision, H.Z. and J.H.; funding acquisition, B.F. and J.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported in part by the National Natural Science Foundation of China (62107027, 62177032) and China Postdoctoral Science Foundation (No. 2021M692006).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available in this article.

**Acknowledgments:** The authors would like to thank the anonymous reviewers for their constructive comments.

**Conflicts of Interest:** The authors declare no competing financial interests. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; and in the decision to publish the results.

## References

1. Zhou, Y.; Wei, Y. Learning hierarchical spectral–spatial features for hyperspectral image classification. *IEEE Trans. Cybern.* **2015**, *46*, 1667–1678. [[CrossRef](#)] [[PubMed](#)]
2. Luo, F.; Du, B.; Zhang, L.; Zhang, L.; Tao, D. Feature learning using spatial-spectral hypergraph discriminant analysis for hyperspectral image. *IEEE Trans. Cybern.* **2018**, *49*, 2406–2419. [[CrossRef](#)] [[PubMed](#)]
3. Yuan, H.; Tang, Y.Y. Spectral–spatial shared linear regression for hyperspectral image classification. *IEEE Trans. Cybern.* **2016**, *47*, 934–945. [[CrossRef](#)] [[PubMed](#)]
4. Paoletti, M.; Haut, J.; Plaza, J.; Plaza, A. Deep learning classifiers for hyperspectral imaging: A review. *Isprs J. Photogramm. Remote Sens.* **2019**, *158*, 279–317. [[CrossRef](#)]
5. Wang, Q.; Lin, J.; Yuan, Y. Salient band selection for hyperspectral image classification via manifold ranking. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 1279–1289. [[CrossRef](#)]
6. Yin, J.; Wang, Y.; Hu, J. A new dimensionality reduction algorithm for hyperspectral image using evolutionary strategy. *IEEE Trans. Ind. Inform.* **2012**, *8*, 935–943. [[CrossRef](#)]
7. Huang, H.Y.; Kuo, B.C. Double nearest proportion feature extraction for hyperspectral-image classification. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 4034–4046. [[CrossRef](#)]
8. Kuo, B.C.; Li, C.H.; Yang, J.M. Kernel nonparametric weighted feature extraction for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 1139–1155.
9. Benediktsson, J.A.; Palmason, J.A.; Sveinsson, J.R. Classification of hyperspectral data from urban areas based on extended morphological profiles. *IEEE Trans. Geosci. Remote Sens.* **2005**, *43*, 480–491. [[CrossRef](#)]
10. Qian, Y.; Ye, M.; Zhou, J. Hyperspectral image classification based on structured sparse logistic regression and three-dimensional wavelet texture features. *IEEE Trans. Geosci. Remote Sens.* **2012**, *51*, 2276–2291. [[CrossRef](#)]
11. Jia, S.; Shen, L.; Zhu, J.; Li, Q. A 3-D Gabor phase-based coding and matching framework for hyperspectral imagery classification. *IEEE Trans. Cybern.* **2018**, *48*, 1176–1188. [[CrossRef](#)]
12. Tang, Y.Y.; Lu, Y.; Yuan, H. Hyperspectral image classification based on three-dimensional scattering wavelet transform. *IEEE Trans. Geosci. Remote Sens.* **2014**, *53*, 2467–2480. [[CrossRef](#)]
13. Melgani, F.; Bruzzone, L. Classification of hyperspectral remote sensing images with support vector machines. *IEEE Trans. Geosci. Remote Sens.* **2004**, *42*, 1778–1790. [[CrossRef](#)]
14. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
15. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
16. Bertinetto, L.; Valmadre, J.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. Fully-convolutional siamese networks for object tracking. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 11–14 October 2016; pp. 850–865.
17. Zhang, H.; Li, Y. Spectral-spatial classification of hyperspectral imagery based on deep convolutional network. In Proceedings of the 2016 International Conference on Orange Technologies (ICOT), Melbourne, VIC, Australia, 18–20 December 2016; pp. 44–47.
18. 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](#)]
19. Li, W.; Wu, G.; Zhang, F.; Du, Q. Hyperspectral image classification using deep pixel-pair features. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 844–853. [[CrossRef](#)]
20. Tarabalka, Y.; Benediktsson, J.A.; Chanussot, J. Spectral–spatial classification of hyperspectral imagery based on partitioned clustering techniques. *IEEE Trans. Geosci. Remote Sens.* **2009**, *47*, 2973–2987. [[CrossRef](#)]
21. Li, Y.; Zhang, H.; Shen, Q. Spectral–spatial classification of hyperspectral imagery with 3D convolutional neural network. *Remote Sens.* **2017**, *9*, 67. [[CrossRef](#)]
22. Zhong, Z.; Li, J.; Luo, Z.; Chapman, M. Spectral–spatial residual network for hyperspectral image classification: A 3-D deep learning framework. *IEEE Trans. Geosci. Remote Sens.* **2017**, *56*, 847–858. [[CrossRef](#)]
23. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

24. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [[CrossRef](#)]
25. Xie, S.; Sun, C.; Huang, J.; Tu, Z.; Murphy, K. Rethinking spatiotemporal feature learning for video understanding. *arXiv* **2017**, arXiv:1712.04851.
26. Lee, H.; Kwon, H. Going deeper with contextual CNN for hyperspectral image classification. *IEEE Trans. Image Process.* **2017**, *26*, 4843–4855. [[CrossRef](#)]
27. Zhang, H.; Li, Y.; Zhang, Y.; Shen, Q. Spectral-spatial classification of hyperspectral imagery using a dual-channel convolutional neural network. *Remote Sens. Lett.* **2017**, *8*, 438–447. [[CrossRef](#)]
28. Makantasis, K.; Karantzalos, K.; Doulamis, A.; Doulamis, N. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 4959–4962.
29. Fang, B.; Li, Y.; Zhang, H.; Chan, J.C.W. Hyperspectral images classification based on dense convolutional networks with spectral-wise attention mechanism. *Remote Sens.* **2019**, *11*, 159. [[CrossRef](#)]
30. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
31. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv* **2017**, arXiv:1704.04861.
32. Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient CNN architecture design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 116–131.
33. Ioffe, S.; Szegedy, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 448–456.
34. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
35. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
36. Xiong, Z.; Yuan, Y.; Wang, Q. AI-NET: Attention inception neural networks for hyperspectral image classification. In Proceedings of the IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 2647–2650.
37. Ruiz Hidalgo, D.; Bacca Cortés, B.; Caicedo Bravo, E. Data classification of hyperspectral images based on inception networks and extended attribute profiles. *Int. J. Remote Sens.* **2020**, *41*, 8717–8738. [[CrossRef](#)]
38. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
39. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; others. Searching for mobilenetv3. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 1314–1324.
40. Fang, B.; Li, Y.; Zhang, H.; Chan, J.C.W. Collaborative learning of lightweight convolutional neural network and deep clustering for hyperspectral image semi-supervised classification with limited training samples. *Isprs J. Photogramm. Remote Sens.* **2020**, *161*, 164–178. [[CrossRef](#)]
41. Li, K.; Ma, Z.; Xu, L.; Chen, Y.; Ma, Y.; Wu, W.; Wang, F.; Liu, Z. Depthwise separable ResNet in the MAP framework for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2020**, *19*, 1–5. [[CrossRef](#)]
42. Meng, Z.; Jiao, L.; Liang, M.; Zhao, F. A lightweight spectral-spatial convolution module for hyperspectral image classification. *IEEE Geosci. Remote Sens. Lett.* **2021**, *19*, 1–5. [[CrossRef](#)]
43. Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **2009**, *22*, 1345–1359. [[CrossRef](#)]
44. Quattoni, A.; Collins, M.; Darrell, T. Transfer learning for image classification with sparse prototype representations. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
45. Yang, J.; Zhao, Y.Q.; Chan, J.C.W. Learning and transferring deep joint spectral-spatial features for hyperspectral classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 4729–4742. [[CrossRef](#)]
46. Lin, J.; Ward, R.; Wang, Z.J. Deep transfer learning for hyperspectral image classification. In Proceedings of the 2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSp), Vancouver, BC, Canada, 29–31 August 2018; pp. 1–5.
47. He, X.; Chen, Y.; Ghamisi, P. Heterogeneous transfer learning for hyperspectral image classification based on convolutional neural network. *IEEE Trans. Geosci. Remote Sens.* **2019**, *58*, 3246–3263. [[CrossRef](#)]
48. Zhao, X.; Liang, Y.; Guo, A.J.; Zhu, F. Classification of small-scale hyperspectral images with multi-source deep transfer learning. *Remote Sens. Lett.* **2020**, *11*, 303–312. [[CrossRef](#)]
49. 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](#)]
50. de Brébisson, A.; Vincent, P. An exploration of softmax alternatives belonging to the spherical loss family. *arXiv* **2015**, arXiv:1511.05042.

51. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2012; Volume 25.
52. Cao, X.; Xu, L.; Meng, D.; Zhao, Q.; Xu, Z. Integration of 3-dimensional discrete wavelet transform and Markov random field for hyperspectral image classification. *Neurocomputing* **2017**, *226*, 90–100. [[CrossRef](#)]
53. Thompson, W.D.; Walter, S.D. A reappraisal of the kappa coefficient. *J. Clin. Epidemiol.* **1988**, *41*, 949–958. [[CrossRef](#)]
54. Hänsch, R.; Ley, A.; Hellwich, O. Correct and still wrong: The relationship between sampling strategies and the estimation of the generalization error. In Proceedings of the 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Fort Worth, TX, USA, 23–28 July 2017; pp. 3672–3675.
55. Xue, Z.; Zhang, M.; Liu, Y.; Du, P. Attention-based second-order pooling network for hyperspectral image classification. *IEEE Trans. Geosci. Remote Sens.* **2021**, *59*, 9600–9615. [[CrossRef](#)]