International Journal of Geo-Information

MDPI

*Article*

# Multisource Hyperspectral and LiDAR Data Fusion for Urban Land-Use Mapping based on a Modified Two-Branch Convolutional Neural Network

**Quanlong Feng [1,2], Dehai Zhu [2,*], Jianyu Yang [2] and Baoguo Li [2]**

[1] College of Resources and Environmental Sciences, China Agricultural University, Beijing 100094, China; fengql@cau.edu.cn

[2] College of Land Science and Technology, China Agricultural University, Beijing 100083, China; ycjyyang@126.com (J.Y.); libg@cau.edu.cn (B.L.)

* Correspondence: zhudehai@263.net; Tel.: +86-10-62737554

Received: 4 November 2018; Accepted: 9 January 2019; Published: 14 January 2019

**Abstract:** Accurate urban land-use mapping is a challenging task in the remote-sensing field. With the availability of diverse remote sensors, synthetic use and integration of multisource data provides an opportunity for improving urban land-use classification accuracy. Neural networks for Deep Learning have achieved very promising results in computer-vision tasks, such as image classification and object detection. However, the problem of designing an effective deep-learning model for the fusion of multisource remote-sensing data still remains. To tackle this issue, this paper proposes a modified two-branch convolutional neural network for the adaptive fusion of hyperspectral imagery (HSI) and Light Detection and Ranging (LiDAR) data. Specifically, the proposed model consists of a HSI branch and a LiDAR branch, sharing the same network structure to reduce the time cost of network design. A residual block is utilized in each branch to extract hierarchical, parallel, and multiscale features. An adaptive-feature fusion module is proposed to integrate HSI and LiDAR features in a more reasonable and natural way (based on "Squeeze-and-Excitation Networks"). Experiments indicate that the proposed two-branch network shows good performance, with an overall accuracy of almost 92%. Compared with single-source data, the introduction of multisource data improves accuracy by at least 8%. The adaptive fusion model can also increase classification accuracy by more than 3% when compared with the feature-stacking method (simple concatenation). The results demonstrate that the proposed network can effectively extract and fuse features for a better urban land-use mapping accuracy.

**Keywords:** convolutional neural networks; multisource data; feature fusion; urban land-use mapping

## 1. Introduction

Urban land-use mapping is of great significance for various urban applications, such as urban planning and designing, urban-environment monitoring, and urban-land surveys [1,2]. Traditional methods for urban land-use mapping are based on the visual interpretation of high-resolution optical remote-sensing imagery and field surveys, which can be quite time-consuming and laborious. Therefore, it is very important to investigate the automatic classification methods for fragmented and complex urban land-use types.

With the development of remote-sensing technology, some researchers started to use multispectral optical imagery and machine-learning methods to automatically extract urban and-cover and land-use information [3–6]. For instance, Lu et al. [3] combined textural and spectral images

with the traditional supervised classification method for urban land-cover classification based on multispectral QuickBird remote-sensing data. Powell et al. [4] utilized spectral-mixture analysis for subpixel urban land-cover mapping based on Landsat imagery. Pu et al. [5] adopted an object-based method and IKONOS imagery for urban land-cover classification. However, due to the complicated composition of urban landscapes and the low spectral resolution of multispectral remote-sensing data, it is very difficult to yield very high classification accuracy. Compared with multispectral remote sensing, hyperspectral remote sensing can obtain hundreds of narrow contiguous spectral bands, which is capable of separating objects with subtle spectral differences. Recent studies also show the great potential of hyperspectral remote sensing in the differentiation of complex urban land-use types [6,7]. Demarchi et al. [6] used APEX 288-band hyperspectral data for urban land-cover mapping based on unsupervised dimensionality-reduction techniques and several machine-learning classifiers. Tong et al. [7] discussed which features of airborne hyperspectral data to use for urban land-cover classification and showed that the synthetic use of shape, texture and spectral information can improve the classification accuracy.

Meanwhile, due to the availability of diverse remote sensors, researchers began to integrate multisource and multisensor data for better characterization of the land surface [8–13]. Since then, the combined use of HSI and Light Detection and Ranging (LiDAR) data has been an active topic [8–13]. The addition of LiDAR data can provide detailed height and shape information of the scene, which can improve classification accuracy when compared with the use of hyperspectral data alone. For instance, roofs and roads that are both made of concrete are difficult to distinguish in hyperspectral images, but they can easily be separated using LiDAR-derived height information due to the significant difference in altitude. Based on the above points, researchers investigated the fusion methods of multisource hyperspectral and LiDAR data. Debes et al. [8] highlighted two methods for hyperspectral and LiDAR data fusion, including a combined unsupervised and supervised classification scheme, and a graph-based method for the fusion of spectral, spatial, and elevation information. Man et al. [9] discussed both the pixel- and feature-level fusion of hyperspectral and LiDAR data for urban land-use classification and showed that the combination of pixel- and object-based classifiers can increase the classification precision. Moreover, the fusion of hyperspectral and LiDAR data has also been applied in many other fields, such as forest monitoring [10,11], volcano mapping [12], and crop-species classification [13].

As for the approaches of multisource remote sensing data fusion, the widely used methods mainly include feature-level fusion and decision-level fusion. Specifically, in the procedure of feature-level fusion, remote sensing data from different sources are firstly processed to extract the relevant features and then fused through feature stacking or feature reconstructing. Man et al. [9] stacked LiDAR features, i.e., nDSM, Intensity and HSI features, i.e., spectral indices and textures to improve the performance of urban land-use classification. Gonzalez et al. [14] also stacked multisource features from color infrared imagery and LiDAR data for object-based mapping of forest habitats. Similar feature stacking approaches can be also found in the studies of Sankey et al. [11] and Sasaki et al. [15]. Different from above studies, Debes et al [8] utilized a fusion graph to project all the original multisource features into a low-dimensional subspace to increase the robustness of the fused features. Compared with feature-level fusion, multisource datasets are separately classified and then fused or integrated in the process of decision-level fusion to generate the final classification results. Sturari et al. [16] proposed a decision-level fusion method for the fusion of LiDAR and multispectral optical data, where the LiDAR classified objects were used as a posteriori in the object rule-based winner-takes-all fusion step.

In addition, all the above studies are based on shallow architectures and hand-crafted feature descriptors, which cannot obtain the fine and abstract high-level features of a complex urban landscape. Deep learning, on the other hand, is capable of modeling high-level feature representations through a hierarchical learning framework [17]. Abstract and invariant features, together with the classifiers, can be simultaneously learned with a multilayer cascaded deep neural network, which outperforms hand-crafted shallow features in computer-vision tasks, such as image classification [18,19], object detection [20], and landmark detection [21,22]. Deep-learning methods

have also been a hot topic in remote sensing [23], and have been successfully applied to building and road extraction [24], wetland mapping [25], cloud detection [26], and land-cover classification [27].

Recently, researchers have started to utilize deep learning for multisource remote-sensing data fusion [28–30]. A typical framework for multisource data fusion based on deep learning is to construct a two-branch network [28–30]. Features from different data sources are first separately extracted via each branch and then fused through feature stacking or feature concatenation. The fused features are passed to the classifier layer to generate the final classification results. For instance, Xu et al. [28] proposed a two-branch convolutional neural network (CNN) for multisource remote-sensing data classification, and the network can achieve better classification performance than existing methods. Huang et al. [29] used a two-branch CNN for extracting both spatial and spectral features of urban land objects for improved urban land-use mapping performance. Hughes et al. [30] adopted a pseudo-Siamese CNN, which also had a two-branch structure, to identify corresponding patches in SAR (Synthetic Aperture Radar) and optical images.

Nevertheless, the above studies that use a two-branch network have two drawbacks that could be improved. Firstly, the data-fusion method of simply stacking or concatenating different features does not consider the importance or contribution of each feature to the final classification task, which could be improved by assigning a specific weight to each feature. Secondly, the backbone of the network is conventional, e.g., AlexNet [18], which can be replaced by other recent network structures.

To tackle these problems, this paper modified the original two-branch neural network [28] to adaptively fuse hyperspectral and LiDAR data for urban land-use classification. The proposed model mainly consists of three parts, i.e., the hyperspectral-imagery (HSI) branch for spatial–spectral feature extraction, the LiDAR branch for height-relevant feature extraction, and a fusion module for the adaptive feature fusion of the two branches. Specifically, the HSI branch and LiDAR branch share the same network structure, which is based on the cascade of a new multiscale residual block in order to reduce the burden of network design. During the training procedure, each branch is first separately trained, and the whole network is then fine-tuned based on each trained branch.

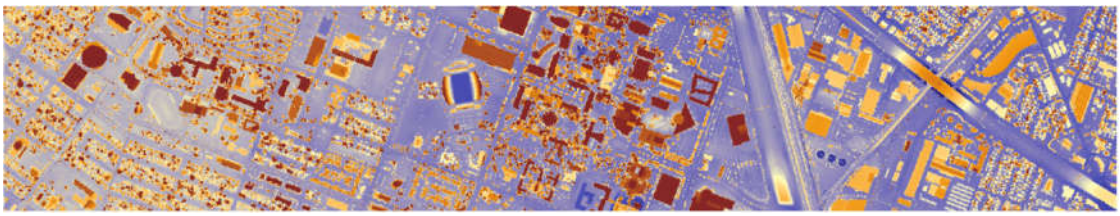
The rest of the paper is organized as follows. Section 2 introduces the study area and dataset. Section 3 presents the detailed architecture of the modified two-branch network. Section 4 shows the experimental results and discussion, and Section 5 provides the main concluding remarks.

## 2. Study Area and Dataset

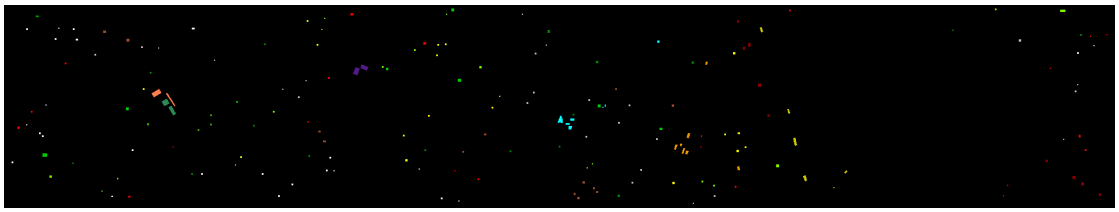
The study area was the University of Houston campus and its neighboring urban areas, which are located in the southeast of Texas, United States. The hyperspectral and LiDAR data were from the 2013 IEEE (Institute of Electrical and Electronic Engineers) GRSS (Geoscience and Remote Sensing Society) Data Fusion Contest [8]. Specifically, the hyperspectral imagery was acquired on 23 June 2012, which consisted of 144 spectral bands ranging from 380 to 1050 nm, with a spectral resolution of 4.8 nm. The spatial resolution was 2.5 m, while the height and width were 349 and 1905 m, respectively.

The LiDAR data were acquired on 22 June 2012 and had already been co-registered with hyperspectral imagery. The spatial resolution of the LiDAR-derived DSM (Digital Surface Model) was also 2.5 m. Figure 1 shows a true-color composite representation of the hyperspectral imagery and the corresponding LiDAR-derived DSM.

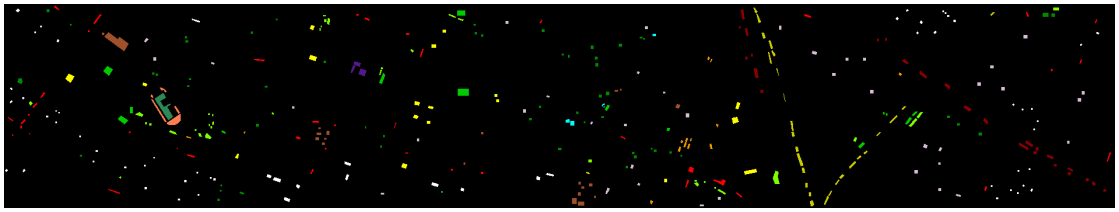
**(a) Hyperspectral imagery.**



**(b) LiDAR-derived DSM.**



**(c) Training samples.**



**(d) Testing samples.**

**Figure 1.** Datasets, training, and testing samples used in this study.

All the training and testing samples are from the Data Fusion Contest. The spatial distribution of training and testing samples are depicted in Figure 1c and d, respectively. There are 15 classes of interest in this study: grass-healthy, grass-stressed, grass-synthetic, tree, soil, water, residential, commercial, road, highway, railway, parking lot 1, parking lot 2, tennis court, and running track. It should be noted that parking lot 1 includes parking garages at both the ground level and elevated areas, while parking lot 2 corresponds to parked vehicles.

The numbers of training and testing samples together with colors for each class are shown in Table 1. As can been seen, the number of training samples is quite limited, which makes it very difficult to achieve high classification accuracy.

**Table 1.** Details of training and testing samples for classification.

| No. | Class name | Training set | Testing set | Color |
|---|---|---|---|---|
| 1 | Healthy grass | 198 | 1053 | |
| 2 | Stressed grass | 190 | 1064 | |
| 3 | Synthetic grass | 192 | 505 | |
| 4 | Tree | 188 | 1056 | |
| 5 | Soil | 186 | 1056 | |
| 6 | Water | 182 | 143 | |
| 7 | Residential | 196 | 1072 | |
| 8 | Commercial | 191 | 1053 | |
| 9 | Road | 193 | 1059 | |

| | | | | |
|---|---|---|---|---|
| 10 | Highway | 191 | 1036 | |
| 11 | Railway | 181 | 1054 | |
| 12 | Parking lot 1 | 192 | 1041 | |
| 13 | Parking lot 2 | 184 | 285 | |
| 14 | Tennis court | 181 | 247 | |
| 15 | Running track | 187 | 473 | |
| | Total | 2832 | 12,197 | |

## 3. Methods

### *3.1. Overall Workflow*

The architecture of the proposed modified two-branch neural network is depicted in Figure 2, which consists of the hyperspectral branch for the spatial–spectral feature extraction and the LiDAR branch for height-relevant feature extraction. The feature-fusion module was utilized to adaptively fuse the features from each branch, and the class label was determined after the fully connected (FC) layer and the softmax classifier.

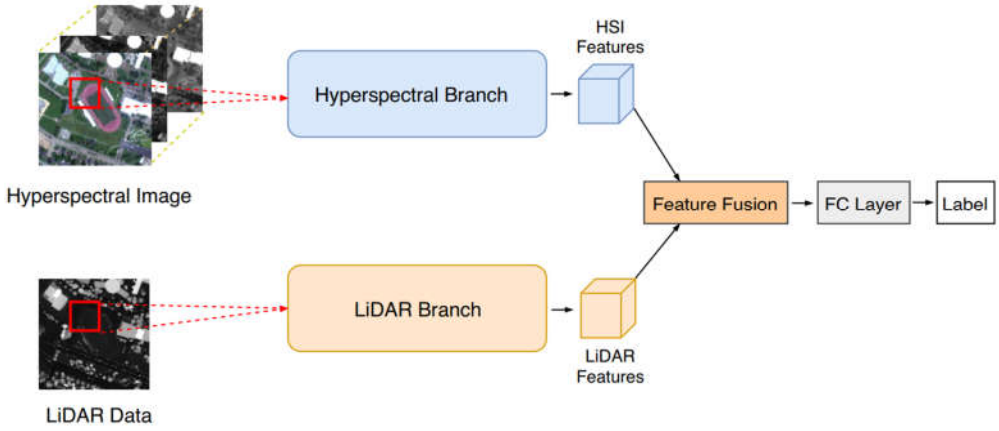



**Figure 2.** Architecture of the proposed two-branch convolutional neural network.

### *3.2. Hyperspectral Branch*

The architecture of the proposed hyperspectral branch is depicted in Figure 3. The input of the HSI branch is a square patch centered at the pij pixel with a side length k. Since the hyperspectral data have 144 bands, some of which are highly corelated, we adopted Principle Component Analysis (PCA) and selected the first 10 components, which accounts for more than 99% of the total variances. Therefore, the input of the HSI branch was a patch with a size of k × k × 10.

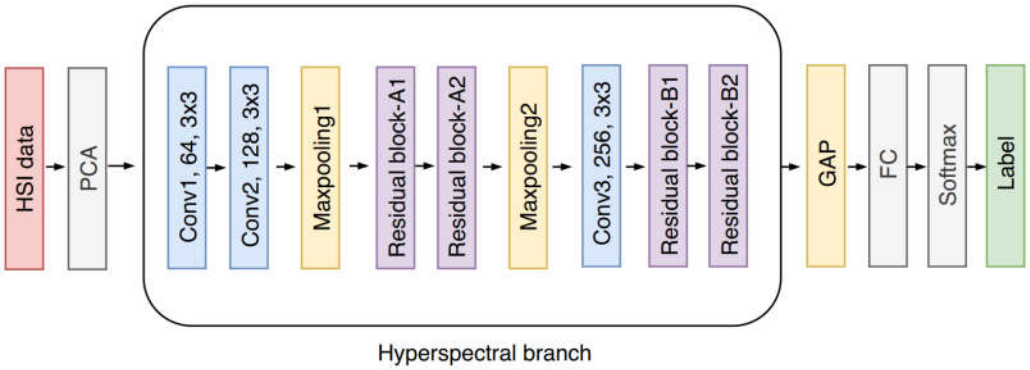



**Figure 3.** Architecture of the proposed hyperspectral imagery (HSI) branch.

The input and output size of each layer of the HSI branch is illustrated in Table 2.

**Table 2.** Configuration of the proposed HSI branch.

| Layer name | Input size | Output size |
|---|---|---|
| Input | 11 × 11 × 10 | - |
| Conv1 | 11 × 11 × 10 | 11 × 11 × 64 |
| Conv2 | 11 × 11 × 64 | 11 × 11 × 128 |
| Maxpooling1 | 11 × 11 × 128 | 6 × 6 × 128 |
| Residual block-A1 | 6 × 6 × 128 | 6 × 6 × 128 |
| Residual block-A2 | 6 × 6 × 128 | 6 × 6 × 128 |
| Maxpooling2 | 6 × 6 × 128 | 3 × 3 × 128 |
| Conv3 | 3 × 3 × 128 | 3 × 3 × 256 |
| Residual block-B1 | 3 × 3 × 256 | 3 × 3 × 256 |
| Residual block-B2 | 3 × 3 × 256 | 3 × 3 × 256 |
| GAP | 3 × 3 × 256 | 1 × 1 × 256 |
| FC | 1 × 1 × 256 | 1 × 1 × 128 |
| Softmax | 1 × 1 × 128 | 1 × 1 × 15 |

As depicted in Figure 3 and Table 2, the proposed HSI branch mainly consists of three convolutional blocks and two maxpooling layers. The first convolutional block includes two convolutional layers, i.e., Conv1 and Conv2, which have 64 and 128 filters, respectively. The second and third convolutional blocks both consist of two residual blocks, i.e., Residual block-A and Residual block-B, whose structure and parameters are shown in Figure 4. Meanwhile, an additional convolutional layer, Conv3, was utilized to increase the dimension of the feature map from the second Residual block-A, which is 128, to match the dimension of the input of the first Residual block-B, which is 256.

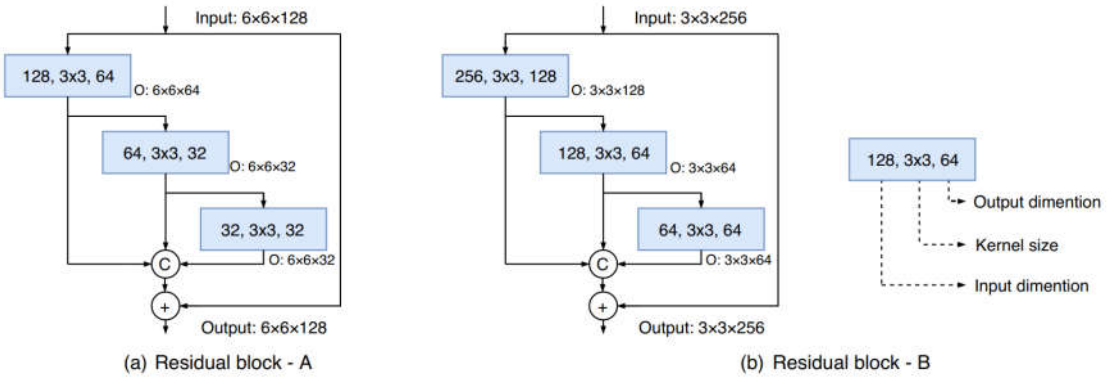


**Figure 4.** Architecture of Residual block-A and Residual block-B in the HSI branch. O: Output; C: Concatenate; ⊕: Sum.

As can be seen from Figure 4, the input dimensions of Residual block-A and B are 6 × 6 × 128 and 3 × 3 × 256, respectively, while the output dimensions are the same as that of the input. The output dimensions of each convolutional layer are also depicted in Figure 4. As for the design of Residual block-A and -B, we referred to the hierarchical, parallel, and multiscale residual block proposed by Bulat [21], which can increase the receptive field size and improve gradient flow at the same time. The residual block shown in Figure 4 has already been successfully used in the field of face landmark detections and showed state-of-the-art performance [21]. The usage of two cascaded residual blocks were inspired by the Face Alignment Network (FAN) [22], which adopted cascaded Hour Glass networks for the extraction of more robust and representative features. Therefore, in this study, we also cascaded the two residual blocks to increase its capability for extracting robust and multiscale features from origin pixel values. Meanwhile, to reduce the risk of overfitting, we utilized L2 regularization for the parameters of all the convolutional layers of the HSI branch. Batch Normalization (BN) [31] was also used after each convolutional layer for a stable training process and to avoid overfitting at the same time.

In addition, the determination of optimum patch size k is very important. A series of experiments were done with different patch sizes, ranging from 9 to 29, according to the spatial resolution of the data and the size of the interested objects. The following figure illustrates the curve of patch size k versus overall accuracy. As depicted in Figure 5, the overall accuracy achieved the highest value of 91.87% when k = 11. When the patch size was larger than 11, the overall accuracy decreased with some fluctuations. This is mainly because a larger patch size could bring more noise than contextual information for the convolutional neural network. Another demerit of a larger patch size is that it could result in the undersegmentation of the remote-sensing data.

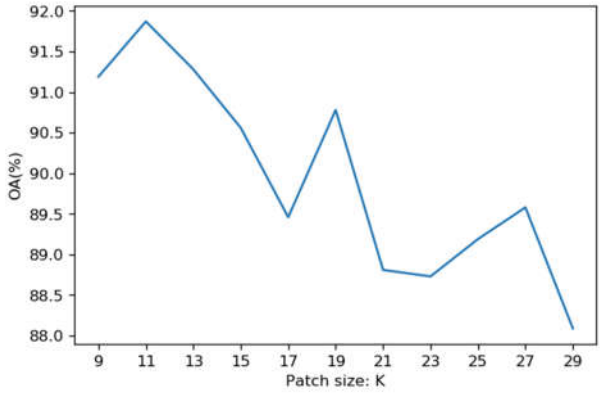



**Figure 5.** Patch size k vs. overall accuracy.

### *3.3. LiDAR Branch*

The input of the LiDAR branch is also a square patch centered at the pij pixel with a side length k = 11. Inspired by the bilinear network [32], which utilized two identical neural networks to learn features with different scales in the field of fine-grained classification, we made the LiDAR branch share the same network structure as the HSI branch. In fact, designing two separate networks could significantly increase the time cost. By unified-designing the HSI and LiDAR branches, it is now quite fast and convenient to formulate the final two-branch neural network. As the LiDAR branch already consisted of cascaded residual blocks, it could also extract the robust and multiscale features from the LiDAR-derived DSM data as expected.

### *3.4. Squeeze-and-Excitation Module for Adaptive Feature Fusion*

Feature-level fusion should be conducted after the extraction of spatial–spectral features and height-relevant features from the HSI branch and the LiDAR branch, respectively. Feature stacking or feature concatenation is often used as the feature-fusion method in previous studies. However, the method of simply stacking different features together does not consider the importance or contribution of each feature to the final classification task, which could be improved by assigning a specific weight to each feature. Inspired by Squeeze-and-Excitation Networks (SENet) [19], which ranked first in the image-classification task of the 2017 ImageNet Large-Scale Visual-Recognition Challenge (ILSVRC), we utilized the Squeeze-and-Excitation module to adaptively learn a specific weight for each feature, thus achieving a more robust and effective feature-fusion method than feature stacking. The architecture of the feature-fusion module is illustrated in Figure 6.

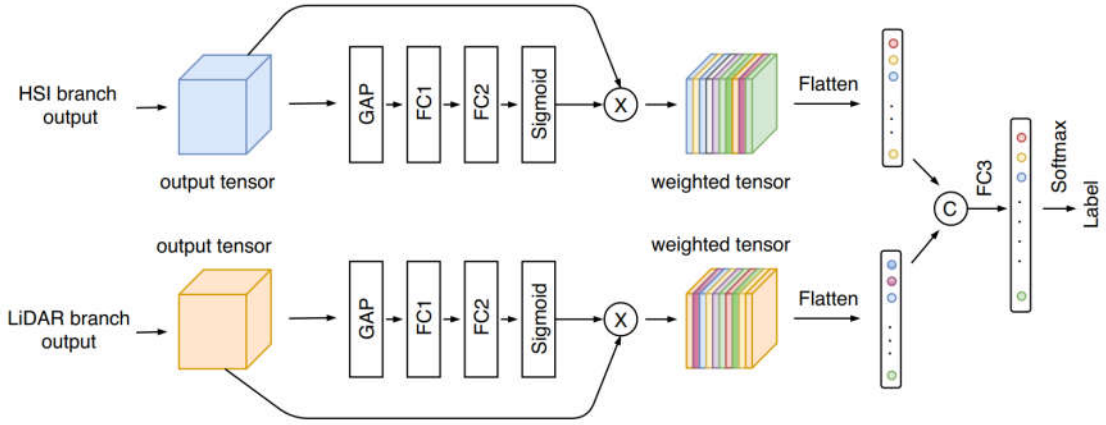



**Figure 6.** Structure of the adaptive feature-fusion module. C: Concatenate; ×: pointwise production.

The configuration of the feature-fusion module is depicted in Table 3, which shows the input and output size of each layer.

**Table 3.** Configuration of the adaptive feature-fusion module.

| Layer name | Input size | Output size |
|---|---|---|
| HSI branch output | 3 × 3 × 256 | - |
| Lidar branch output | 3 × 3 × 256 | - |
| GAP | 3 × 3 × 256 | 1 × 1 × 256 |
| FC1 | 1 × 1 × 256 | 1 × 1 × 64 |
| FC2 | 1 × 1 × 64 | 1 × 1 × 256 |
| Sigmoid | 1 × 1 × 256 | 1 × 1 × 256 |
| Flatten | 3 × 3 × 256 | 1 × 1 × 2304 |
| Concat | 1 × 1 × 2304, 1 × 1 × 2304 | 1 × 1 × 4608 |
| FC3 | 1 × 1 × 4608 | 1 × 1 × 128 |
| Softmax | 1 × 1 × 128 | 1 × 1 × 15 |

As shown in Figure 6 and Table 3, the input features of the fusion module are extracted from the HSI and the LiDAR branches, both of which have the same dimension of 3 × 3 × 256. The original features are separately passed through two parallel SE blocks, after which they are recalibrated or reweighted. As can be seen, the recalibrated features from both branches still have the same dimensions of 3 × 3 × 256. Afterward, they are flattened to the dimensions of 1 × 1 × 2304, respectively, and then concatenated to generate a feature vector with dimensions of 1 × 1 × 4608. Next, the concatenated vector is passed through a fully connected layer with 128 neurons and finally fed into the softmax layer to generate the predicted class labels.

The structure of the SE block is described as follows. For any given features U, they are first passed through a global average pooling (GAP) layer to produce a channel descriptor, which embeds

the global distribution of channel-wise feature responses. This is followed by two FC layers and a sigmoid layer, in which the channel-specific weight can be learned through a self-gating mechanism based on channel dependence. The output features of the SE block were already recalibrated or reweighted, leading to adaptively emphasizing the informative features and suppressing the less useful ones. Compared with the traditional feature-stacking fusion method, the SE module in this paper can provide a more effective and rational way for feature-level fusion of multisource data.

### *3.5. Data Augmentation and Network Training*

As is known, training a deep CNN model needs a large quantity of labeled data. However, for remote-sensing applications, it is laborious and time-consuming to obtain enough labeled data. To address this issue, data augmentation was utilized in this study. Original training patches were rotated 90°, 180° and 270°, flipped left and right, up, and down to increase the number of training samples. Furthermore, classes with fewer training samples were oversampled to tackle the problem of class imbalance.

All the parameters of the proposed two-branch network need to be trained to generate the best model for urban land-use classification. In this study, a two-step training strategy was used to train the whole network. Firstly, the HSI and LiDAR branches were separately trained with a larger initial learning rate of $10^{-4}$. Secondly, the pretrained HSI and LiDAR branches were merged through the adaptive feature-fusion module, and the whole network was fine-tuned with a smaller initial learning rate of $10^{-5}$. The Adam optimizer [33] was used due to its capability of automatically adjusting the learning rate, which could result in a faster and more stable training procedure.

Focal loss [20] was utilized as the loss function in this study instead of traditional cross-entropy loss. This is mainly because focal loss has the merits of downweighing the loss assigned to well-classified examples, which prevents the vast number of easy examples from overwhelming the classifier during training [20].

In this study, 90% of the training set was randomly selected to optimize the parameters of the proposed two-branch neural network. The remaining 10% of the training samples were used as the validation set to justify the performance of the network during training process. As for the testing set, it was only used for calculating the final overall accuracy and the confusion matrix after the network was well-trained.

The proposed two-branch network was trained with the TensorFlow library [34] on Ubuntu 16.04 operation system with an Intel CORE i7-7800 @ 3.5 GHz CPU and an NVIDIA GTX TitanX GPU with 12 GB memory.

### *3.6. Accuracy Assessment*

In order to assess the performance of the proposed two-branch network for urban land-use classification, both visual evaluation and a confusion matrix were adopted in this study. Visual inspection was used to check the visual effects, while the confusion matrix, derived from the testing samples, was used to quantitatively assess the classification accuracy of the proposed method. It should be noted that all the testing samples are from the 2013 IEEE GRSS Data Fusion Contest, which are the same as in Reference [28].

## **4. Results and Discussion**

### *4.1. Results of Urban Land-Use Classification*

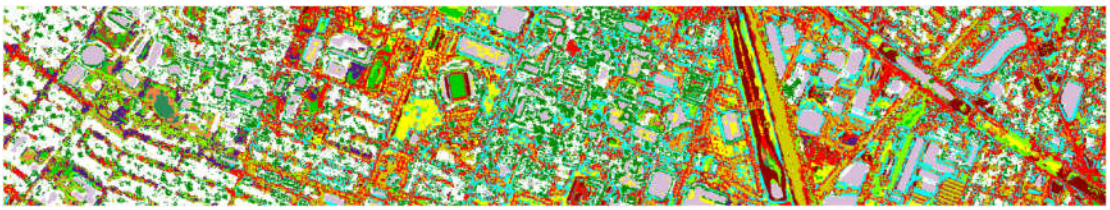
In order to evaluate the performance of the proposed two-branch neural network for urban land-use mapping, a series of classification maps are depicted in Figure 7 including the following cases:

(a) HSI branch only, i.e., using only HSI data and the HSI branch for classification;

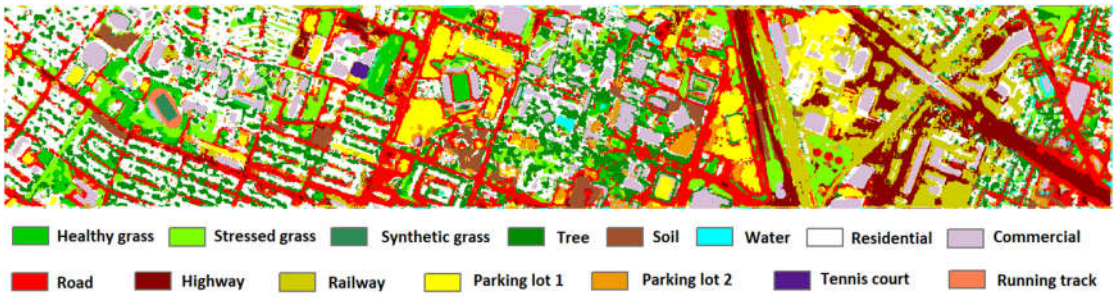
(b) LiDAR branch only, i.e., using only LiDAR data and the LiDAR branch for classification;

(c) the proposed two-branch CNN.

**(a) HSI branch only.**



**(b) LiDAR branch only.**




**(c) Proposed two-branch convolutional neural network (CNN).**

**Figure 7.** Classification maps for (**a**) HSI branch only; (**b**) LiDAR branch only; (**c**) proposed two-branch CNN.

It is evident that the synthetic use of HSI and LiDAR data leads to a classification map with a better visual effect and higher quality when compared with the results of only the HSI branch and only the LiDAR branch.

Meanwhile, the HSI branch yields a better classification map with fewer errors than that of the LiDAR branch. However, due to the large spectral variance of different urban land-use types, hyperspectral data alone could also result in inaccurate classification results. For instance, the eastern area of the image is covered by some clouds, which leads to the spectral distortion of certain land-use types, resulting in more classification errors. The LiDAR data alone also do not contain enough information to differentiate complicated urban objects, especially for different objects with the same or similar elevation. Nonetheless, the fusion of HSI and LiDAR data can avoid the above demerits and benefit from both the spectral characteristics of the HSI image and the geometric information of the LiDAR data, which could lead to a better classification map.

### *4.2. Accuracy-Assessment Results*

In order to quantitatively evaluate the proposed approach in the study, the confusion matrix, together with the overall accuracy (OA) and Kappa coefficient, were calculated based on the testing samples. Results are shown in the Table 4.

**Table 4.** Confusion matrix of the proposed method.

| | **Testing data** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | UA |

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 875 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 100 |
| 2 | 0 | 894 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 100 |
| 3 | 0 | 0 | 504 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 100 |
| 4 | 15 | 0 | 0 | 1020 | 2 | 0 | 9 | 6 | 5 | 0 | 0 | 3 | 0 | 0 | 0 | 96.2 |
| 5 | 0 | 0 | 0 | 0 | 1051 | 0 | 0 | 0 | 10 | 0 | 0 | 0 | 0 | 0 | 0 | 99.0 |
| 6 | 0 | 0 | 0 | 0 | 1 | 143 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 99.3 |
| 7 | 9 | 0 | 0 | 32 | 1 | 0 | 968 | 8 | 12 | 22 | 0 | 5 | 0 | 0 | 0 | 91.6 |
| 8 | 0 | 11 | 0 | 0 | 0 | 0 | 14 | 989 | 0 | 97 | 0 | 5 | 0 | 0 | 0 | 88.6 |
| 9 | 0 | 0 | 0 | 2 | 0 | 0 | 21 | 0 | 904 | 2 | 5 | 5 | 0 | 0 | 0 | 96.3 |
| 10 | 80 | 6 | 0 | 2 | 0 | 0 | 0 | 0 | 66 | 838 | 1 | 0 | 8 | 0 | 0 | 83.7 |
| 11 | 74 | 152 | 0 | 0 | 0 | 0 | 57 | 32 | 13 | 77 | 1020 | 0 | 12 | 0 | 0 | 71.0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 11 | 47 | 0 | 21 | 1014 | 0 | 0 | 0 | 92.8 |
| 13 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 5 | 2 | 0 | 7 | 9 | 265 | 0 | 0 | 91.4 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 247 | 0 | 99.6 |
| 15 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 473 | 99.6 |
| PA | 83.1 | 84.0 | 99.8 | 96.6 | 99.5 | 100 | 90.3 | 93.9 | 85.4 | 80.9 | 96.8 | 97.4 | 93.0 | 100 | 90.3 | |
| OA | 91.87 | | | Kappa | | 0.9117 | | | | | | | | | | |

1: Healthy grass; 2: Stressed grass; 3: Synthetic grass; 4: Tree; 5: Soil; 6: Water; 7: Residential; 8: Commercial; 9: Road; 10: Highway; 11: Railway; 12: Parking lot 1; 13: Parking lot 2; 14: Tennis court; 15: Running track.

The proposed two-branch network shows good performance, with an OA of 91.87% and a Kappa of 0.9117. However, the highway class had the lowest producer accuracy (PA) with 80.89%, while all the other classes had a higher PA of more than 83%. This could be due to the spectral mixture between highway and other impervious surface types, such as railway and commercial areas, since they all consist of concrete materials. It should also be noted that all highway training samples are outside the cloud-covered regions, while nearly half of the highway testing samples are from the cloud-covered regions. This could cause the spectral inconsistency between training and testing samples of the highway class, which could result in relatively lower classification accuracy.

In addition, the confusion matrix indicates that most classification errors occurred among the following land-use types: Highway, railway, road, and parking lot 1. This is mainly because all those land-use categories belong to impervious surfaces that share similar spectral properties. Highway, railway, and road also share similar shape features, which could increase the difficulty in separating between them using a patch-based CNN model, since the CNN takes spatial contextual information into consideration when classifying each pixel. Other errors occurred where several healthy-grass and stressed-grass pixels were misclassified as the railway. This is uncommon in remote-sensing image classification, since grass and railway share different spectral characteristics. However, when checking the classification map, it is in the eastern cloud-covered regions that several grass areas were misclassified as railway. This is mainly because the existence of heavy cloud distorted the spectral curves of the grass, which led to the uncommon classification errors.

### *4.3. Ablation Analysis*

To further evaluate the performance of the proposed method, a series of ablation experiments were done including: (a) only the HSI branch, (b) only the LiDAR branch, and (c) feature-stacking (i.e., using stacked or concatenated features of HSI and LiDAR instead of adaptive fusion for classification). The results of class-level classification accuracy are illustrated in Table 5 including all the above three cases together with the proposed two-branch network.

**Table 5.** Class-level classification accuracy.

| No. | Class name | Only HSI branch | Only LiDAR branch | Feature-stacking | Proposed |
|---|---|---|---|---|---|
| 1 | Healthy grass | 82.05 | 47.39 | 82.33 | 83.09 |
| 2 | Stressed grass | 84.02 | 33.46 | 84.96 | 84.02 |
| 3 | Synthetic grass | 96.04 | 91.49 | 99.80 | 99.80 |
| 4 | Tree | 86.27 | 74.53 | 89.58 | 96.59 |
| 5 | Soil | 94.51 | 33.81 | 99.62 | 99.53 |
| 6 | Water | 79.02 | 58.74 | 98.60 | 100.00 |
| 7 | Residential | 95.06 | 57.65 | 83.96 | 90.30 |
| 8 | Commercial | 72.17 | 90.69 | 85.84 | 93.92 |
| 9 | Road | 82.06 | 37.39 | 86.31 | 85.36 |
| 10 | Highway | 65.35 | 42.08 | 69.88 | 80.89 |
| 11 | Railway | 71.82 | 75.90 | 90.61 | 96.77 |
| 12 | Parking lot 1 | 92.03 | 25.46 | 93.56 | 97.41 |
| 13 | Parking lot 2 | 85.61 | 61.40 | 91.58 | 92.98 |
| 14 | Tennis court | 97.17 | 74.49 | 100.00 | 100.00 |
| 15 | Running track | 95.06 | 57.65 | 83.96 | 90.30 |
| | OA (%) | 83.83 | 53.42 | 88.25 | 91.87 |
| | Kappa | 0.8244 | 0.4967 | 0.8725 | 0.9117 |

Table 5 indicates that the LiDAR branch alone achieved the lowest classification accuracy with an OA of 53.42% and a Kappa of 0.4967. This is mainly due to the fact that height information alone can hardly separate different land-use types in complicated urban regions. Meanwhile, the HSI branch alone achieved much higher accuracy than that of only the LiDAR branch, with an OA of 83.83% and a Kappa of 0.8244. The reason why the HSI branch alone outperformed the LiDAR branch alone is that HSI images can provide much more abundant spectral and spatial information of land surfaces than LiDAR-derived DSMs, leading to a higher capability of differentiating complex urban land-use types.

Table 5 also indicates that, when compared with single-source data alone, the integration of multisource HSI and LiDAR data leads to a significant improvement of classification accuracy for almost each urban land-use class. This is reasonable since the separability of urban objects could increase if we simultaneously integrate multiple spectral and elevation features. Compared with hyperspectral data alone, the integration of LiDAR data improved OA by 4.42% and 8.04% through feature-stacking and the proposed two-branch network, respectively. In terms of class-level accuracy, the main contribution of LiDAR data was in the following classes: Synthetic grass, tree, soil, water, commercial, railway, and parking lot 1 and 2. This is due to some of the classes (e.g., grass and tree) sharing very similar spectral characteristics but having different height values; therefore, the inclusion of LiDAR-derived DSMs could significantly improve the separability between these classes.

It should be noted that the proposed two-branch network, which uses adaptive feature fusion, outperforms the traditional feature-stacking method by improving OA from 88.25% to 91.87%, with an increase of 3.62%. This is because, when simply stacking all features together, the values of each feature can be significantly unbalanced, and the information carried by each feature may not be equally represented. Therefore, we introduced the adaptive squeeze-and-excitation module to automatically assign a weight to each feature according to its importance, which could integrate multiple features in a more natural and reasonable way, resulting in the accuracy improvement of 3.62%.

Since the proposed approach used a pixel-centered patch as input to the networks, more comparative experiments should be done to compare the performance between the pixel-based and patch-based classification, and to investigate the effect of the PCA and non-PCA approaches. Therefore, a series of ablation experiments were performed, and the comparison results are shown in the following table.

**Table 6.** Comparison with pixel-based and non-Principle Component Analysis (PCA) methods.

| Method | OA | Kappa |
|---|---|---|
| Pixel-based and non-PCA | 81.49% | 0.8001 |
| Pixel-based and PCA | 86.05% | 0.8486 |
| Patch-based and non-PCA | 89.38% | 0.8849 |
| Patch-based and PCA | 91.87% | 0.9117 |

Specifically, since the input was changed to 1D pixel vectors, all the 2D convolutional layers of the original patch-based two-branch CNN were replaced by 1D convolutional layers in the pixel-based CNN, while all parameters remained the same. As can be seen from Table 6, the patch-based models outperformed the pixel-based ones with an accuracy increase of 7.89% and 5.82% for the non-PCA and PCA-based approaches, respectively. This is mainly because, when compared with the patch-based model, the pixel-based model only considers the spectral characteristics of the land object. However, the patch-based model can take both the spectral and spatial contextual information into account, leading to more discriminative and representative features which are essential for classification. When compared with non-PCA approaches, the usage of PCA has a positive effect on classification, leading to an accuracy increase of 4.56% and 2.49% for the pixel-based and patch-based-models, respectively. This is due to the fact that PCA can effectively reduce the data redundancy of the original hyperspectral imagery, which can reduce the overfitting risk of the convolutional neural network and, thus, improve its generalization ability when predicting on new datasets.

Additionally, the proposed approach can be considered as a reference framework for multisource data fusion in the remote-sensing field.

### *4.4. Comparison with Other Methods*

To further justify the performance of the proposed approach, it should be compared with other widely used machine-learning methods, such as random forest (RF) [35], support vector machines (SVM) [36], and state-of-the-art methods.

For RF, the Gini coefficient was used as the index for feature selection. For SVM, the Radial Basis Function (RBF) was used as the kernel function. As for the determination of the hyperparameters of RF and SVM, we utilized the grid-search method to find the optimal values. Specifically, the ranges of used parameters for RF are as follows. The number of trees ranged from 50 to 500 with a step of 10, while the max depth had a range of 5 to 15 with a step of 2. For SVM, gamma ranged from 0.001 to 0.1 with a step of 0.001, while punishment coefficient C had a range of 10 to 200 with a step of 10. After the procedure of grid search, RF achieved the best overall accuracy of 83.97% when the number of trees was 200 and the max depth was 13. Meanwhile, SVM achieved the best accuracy of 84.16% with a gamma of 0.01 and a C of 100.

Meanwhile, we selected Xu's model [28] as a strong baseline since it first utilized a two-branch CNN for HSI and LiDAR data fusion, and achieved an OA of 87.98%, also on the 2013 IEEE GRSS Data Fusion Contest testing dataset. All the above methods were trained and tested with the same training and testing samples as the proposed method to maintain fairness. The accuracy comparison results are listed in Table 7.

**Table 7.** Accuracy comparison with other methods.

| Method | OA | Kappa |
|---|---|---|
| Random Forest | 83.97% | 0.8264 |
| Support Vector Machine | 84.16% | 0.8282 |
| Xu et al. [28] | 87.98% | 0.8698 |
| Our two-branch CNN | 91.87% | 0.9117 |

Table 7 indicates that our proposed modified two-branch CNN outperformed both RF and SVM with an OA improvement of 7.90% and 7.71%, respectively. This was expected since, when compared

with traditional machine-learning methods, the CNN could learn high-level spatial features of complicated and fragmented urban land-use types, which led to a more robust and accurate classification result.

When compared with Xu's state-of-the-art model, the proposed method in this study improved OA from 87.98% to 91.87%, with an increase of 3.89%. However, when using feature-stacking, the modified two-branch CNN in this study only achieved a slight accuracy increase of 0.27% compared to Xu's model. This indicates that, when compared with the modification of the network structure, the introduction of the feature-fusion module contributed more to the increase of classification accuracy. This is because the feature-fusion module could learn the importance of each feature, which can emphasize more effective features while suppressing the less informative ones, leading to a more reasonable and robust fusion strategy for multisource remote-sensing data. The backbone of the modified two-branch CNN is less effective than the feature-fusion strategy and does not show superior performance to that of Xu's model.

As stated above, the fusion method of this study is more effective than the model structure itself, therefore, a more comprehensive comparison with existing methods is necessary. In fact, as stated in the Introduction, most of the feature-level fusion studies just simply stacked and concatenated the features from LiDAR and HSI data and then carried out the classification based on machine learning classifiers such as decision tree, support vector machine and random forest. Relevant studies include that of Man et al. [9], Gonzalez et al. [14], Sasaki et al. [16]. However, these approaches gave equal importance to all the features, which could bring in redundant information and extra noise. Different from those feature-stacking methods, the feature fusion approach in this study takes the importance of multisource features into account, which could effectively highlight those most informative features while reduce the noisy ones. Meanwhile, some existing methods designed a feature fusion model to reconstruct the multisource features to increase the classification performance. One concrete example is Debes's study [8], in which a graph-based fusion model was used to re-project all the features into a low-dimensional subspace to increase the robustness of the fused features. Actually, the newly reconstructed features were more informative with less noises, however, this method was not that straight-forward when compared with our approach, where all the original features were directly re-weighted in our feature-fusion model. Nonetheless, the graph-based fusion method can be introduced in the deep learning model in future research.

## 5. Conclusions

This paper proposed a modified two-branch convolutional neural network for urban land-use mapping using multisource hyperspectral and LiDAR data. The proposed two-branch network consists of an HSI branch and a LiDAR branch, both of which share the same network structure in order to reduce the burden and time cost of network design. Within the HSI and LiDAR branches, a hierarchical, parallel, and multiscale residual block was utilized, which could simultaneously increase the receptive field size and improve gradient flow. An adaptive feature-fusion module based on a Squeeze-and-Excitation Net was proposed to fuse the HSI and LiDAR features, which could integrate multisource features in a natural and reasonable way. Experiment results showed that the proposed two-branch network had good performance, with an OA of almost 92% on the 2013 IEEE GRSS Data Fusion Contest dataset. When compared with hyperspectral data alone, the introduction of LiDAR data increased OA from almost 84% to 92%, which indicates that the integration of multisource data could improve classification accuracy in complicated urban landscapes. The proposed adaptive fusion method increased accuracy by more than 3% when compared with the traditional feature-stacking method, which justifies its usefulness in multisource data fusion. The two-branch CNN in this paper also outperformed traditional machine-learning methods, such as random forest and support vector machines.

This paper demonstrates that the modified two-branch network can effectively integrate multisource features from hyperspectral and LiDAR data, showing good performance in urban land-use mapping. Future work should be carried out on more datasets to further justify the performance of the proposed method.

**Author Contributions:** Q.F. proposed the modified two-branch convolutional neural network of this study, and contributed to data preprocessing, the experiments, and the writing of the manuscript. D.Z. and J.Y. contributed to the experiment discussion and manuscript revision. B.L. mainly contributed to the manuscript revision.

**Funding:** This study was funded and supported by the China Postdoctoral Science Foundation (2018M641529), and Ministry of Land and Resources Industry Public Welfare projects (201511010-06).

**Acknowledgments:** Special thanks to the anonymous referees and editors for very useful comments and suggestions to help improve the quality of this paper. Besides, the authors would like to thank the committee of the 2013 IEEE GRSS Data Fusion Contest for providing the hyperspectral and LiDAR data, and their efforts in promoting the development of the multisource remote-sensing data fusion.

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

## References

1. Chen, X.; Zhao, H.; Li, P.; Yin, Z. Remote sensing image-based analysis of the relationship between urban heat island and land use/cover changes. *Remote Sens. Environ.* **2006**, *104*, 133–146.
2. Myint, S.W.; Gober, P.; Brazel, A.; Grossman-Clarke, S.; Weng, Q. Per-pixel vs. object-based classification of urban land cover extraction using high spatial resolution imagery. *Remote Sens. Environ.* **2011**, *115*, 1145–1161.
3. Lu, D.; Hetrick, S.; Moran, E. Land Cover Classification in a Complex Urban-Rural Landscape with QuickBird Imagery. *Photogramm Eng. Remote Sens*. **2010**, *10*, 1159–1168.
4. Powell, R.L.; Roberts, D.A.; Dennison, P.E.; Hess, L.L. Sub-pixel mapping of urban land cover using multiple endmember spectral mixture analysis: Manaus, Brazil. *Remote Sens. Environ*. **2007**, *106*, 253–267.
5. Pu, R.; Landry, S.; Yu, Q. Object-based urban detailed land cover classification with high spatial resolution IKONOS imagery. *Int. J. Remote Sens*. **2011**, *32*, 3285–3308.
6. Demarchi, L.; Canters, F.; Cariou, C.; Licciardi, G.; Chan, J.C. Assessing the performance of two unsupervised dimensionality reduction techniques on hyperspectral APEX data for high resolution urban land-cover mapping. *ISPRS J. Photogramm. Remote Sens*. **2014**, *87*, 166–179.
7. Tong, X.; Xie, H.; Weng, Q. Urban Land Cover Classification with Airborne Hyperspectral Data: What Features to Use? *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens*. **2014**, *7*, 3998–4009.
8. Debes, C.; Merentitis, A.; Heremans, R.; Hahn, J.; Frangiadakis, N.; Kasteren, T.V.; Liao, W.; Bellens, R.; Pizurica, A.; Gautama, S.; et al. Hyperspectral and LiDAR Data Fusion: Outcome of the 2013 GRSS Data Fusion Contest. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens*. **2014**, *7*, 2405–2418.
9. Man, Q.; Dong, P.; Guo, H. Pixel- and feature-level fusion of hyperspectral and lidar data for urban land-use classification. *Int. J. Remote Sens*. **2015**, *36*, 1618–1644.
10. Dalponte, M.; Bruzzone, L.; Gianelle, D. Fusion of Hyperspectral and LIDAR Remote Sensing Data for Classification of Complex Forest Areas. *IEEE Trans. Geosci. Remote Sens*. **2008**, *46*, 1416–1427.
11. Sankey, T.; Donager, J.; McVay, J.; Sankey, J.B. UAV lidar and hyperspectral fusion for forest monitoring in the southwestern USA. *Remote Sens. Environ*. **2017**, *195*, 30–43.
12. Kereszturi, G.; Schaefer, L.N.; Schleiffarth, W.K.; Procter, J.; Pullanagari, R.R.; Mead, S.; Kennedy, B. Integrating airborne hyperspectral imagery and LiDAR for volcano mapping and monitoring through image classification. *Int. J. Appl. Earth Obs. Geoinf.* **2018**, *73*, 323–339.
13. Liu, X.; Bo, Y. Object-Based Crop Species Classification Based on the Combination of Airborne Hyperspectral Images and LiDAR Data. *Remote Sens*. **2015**, *7*, 922–950.
14. Gonzalez, R.S.; Latifi, H.; Weinacker, H.; Dees, M., Koch, B.; Heurich, M. Integrating LiDAR and high-resolution imagery for object-based mapping of forest habitats in a heterogeneous temperate forest landscape. *Int. J. Remote Sens.* **2018**, 1–26.
15. Sasaki, T.; Imanishi, J.; Ioki, K.; Morimoto, Y.; Kitada, K. Object-based classification of land cover and tree species by integrating airborne LiDAR and high spatial resolution imagery data. *Landsc. Ecol. Eng.* **2012**, *8*: 157–171.
16. Sturari, M.; Frontoni, E.; Pierdicca, R.; Mancini, A.; Malinverni, E.S.; Tassetti, A.N.; Zingaretti, P. Integrating elevation data and multispectral high-resolution images for an improved hybrid Land Use/Land Cover mapping. *Eur. J. Remote Sens*. **2017**, *50*, 1–17.
17. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.

18. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems (NIPS), Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
19. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018.
20. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 2999–3007.
21. Bulat, A.; Tzimiropoulos, G. Binarized Convolutional Landmark Localizers for Human Pose Estimation and Face Alignment with Limited Resources. In Proceedings of IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 3706–3714.
22. Bulat, A.; Tzimiropoulos, G. How far are we from solving the 2D and 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks). In Proceedings of IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 1021–1030.
23. Zhu, X.; Tuia, D.; Mou, L.; Xia, G.; Zhang, L.; Xu, F.; Fraundorfer, F. Deep Learning in Remote Sensing: A Comprehensive Review and List of Resources. *IEEE Geosci. Remote Sens. Mag*. **2017**, *5*, 8–36.
24. Alshehhi, R.; Marpu, P.R.; Woon, W.L.; Mura, M.D. Simultaneous extraction of roads and buildings in remote sensing imagery with convolutional neural networks. *ISPRS J. Photogramm. Remote Sens*. **2017**, *130*, 139–149.
25. Rezaee, M.; Mahdianpari, M.; Zhang, Y.; Salehi, B. Deep Convolutional Neural Network for Complex Wetland Classification Using Optical Remote Sensing Imagery. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens*. **2018**, *11*, 3030–3039.
26. Chen, Y.; Fan, R.; Bilal, M.; Yang, X.; Wang, J.; Li, W. Multilevel Cloud Detection for High-Resolution Remote Sensing Imagery Using Multiple Convolutional Neural Networks. *ISPRS Int. J. Geo-Inf*. **2018**, *7*, 181.
27. Rußwurm, M.; Körner, M. Multi-Temporal Land Cover Classification with Sequential Recurrent Encoders. *ISPRS Int. J. Geo-Inf*. **2018**, *7*, 129.
28. Xu, X.; Li, W.; Ran, Q.; Du, Q.; Gao, L.; Zhang, B. Multisource Remote Sensing Data Classification Based on Convolutional Neural Network. *IEEE Trans. Geosci. Remote Sens*. **2018**, *56*, 937–949.
29. Huang, B.; Zhao, B.; Song, Y. Urban land-use mapping using a deep convolutional neural network with high spatial resolution multispectral remote sensing imagery. *Remote Sens. Environ*. **2018**, *214*, 73–86.
30. Hughes, L.H.; Schmitt, M.; Mou, L.; Wang, Y.; Zhu, X. Identifying Corresponding Patches in SAR and Optical Images with a Pseudo-Siamese CNN. *IEEE Geosci. Remote Sens. Lett*. **2018**, *15*, 784–788.
31. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.
32. Lin, T.Y.; RoyChowdhury, A.; Maji, S. Bilinear CNN Models for Fine-grained Visual Recognition. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 13–16 December 2015; pp. 1449–1457.
33. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980.
34. TensorFlow. Available online: https://tensorflow.google.cn/ (accessed on 3 November 2018).
35. Breiman, L. Random forests. *Mach. Learn*. **2001**, *45*, 5–32.
36. Chapelle, O.; Vapnik, V.; Bousquet, O.; Mukherjee, S. Choosing Multiple Parameters for Support Vector Machines. *Mach. Learn*. **2002**, *46*, 131–159.