remote sensing

MDPI

Article

Real-Time Dense Semantic Labeling with Dual-Path Framework for High-Resolution Remote Sensing Image

Yuhao Wang [1,2], Chen Chen [3], Meng Ding [4] and Jiangyun Li [1,2,*]

1 School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China; b20140353@xs.ustb.edu.cn
2 Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, Beijing 100083, China
3 Department of Electrical and Computer Engineering, The University of North Carolina at Charlotte, Charlotte, NC 28223, USA; chen.chen@uncc.edu
4 Thermo Fisher Scientific, Richardson, TX 75081, USA; meng.ding@okstate.edu
* Correspondence: leejy@ustb.edu.cn

Received: 7 November 2019; Accepted: 7 December 2019; Published: 14 December 2019

Abstract: Dense semantic labeling plays a pivotal role in high-resolution remote sensing image research. It provides pixel-level classification which is crucial in land cover mapping and urban planning. With the recent success of the convolutional neural network (CNN), accuracy has been greatly improved by previous works. However, most networks boost performance by involving too many parameters and computational overheads, which results in more inference time and hardware resources, while some attempts with light-weight networks do not achieve satisfactory results due to the insufficient feature extraction ability. In this work, we propose an efficient light-weight CNN based on dual-path architecture to address this issue. Our model utilizes three convolution layers as the spatial path to enhance the extraction of spatial information. Meanwhile, we develop the context path with the multi-fiber network (MFNet) followed by the pyramid pooling module (PPM) to obtain a sufficient receptive field. On top of these two paths, we adopt the channel attention block to refine the features from the context path and apply a feature fusion module to combine spatial information with context information. Moreover, a weighted cascade loss function is employed to enhance the learning procedure. With all these components, the performance can be significantly improved. Experiments on the Potsdam and Vaihingen datasets demonstrate that our network performs better than other light-weight networks, even some classic networks. Compared to the state-of-the-art U-Net, our model achieves higher accuracy on the two datasets with 2.5 times less network parameters and 22 times less computational floating point operations (FLOPs).

Keywords: remote sensing image; real-time dense semantic labeling; convolutional neural networks; light-weight

1. Introduction

High-resolution remote sensing images collected by satellites or unmanned drones are adequate to obtain detailed information about the observed surface and have been widely used in various applications, such as land-use analysis, precision agriculture, urban planning, and disaster warning [1,2]. Recent advances in remote sensing technology have significantly increased the availability of high-resolution image [3]. With the support of sufficient high-quality data, dense semantic labeling has been a pivotal research domain in remote sensing [4–7].

Dense semantic labeling is a pixel-level classification task and aims to assign each pixel with a class label of given categories [8–10]. In the past few years, many machine learning approaches have been developed to handle this challenge. Among them, convolutional neural network (CNN) based methods achieved the best performance [11–14]. Unlike the traditional manually designed methods, CNN is driven by data and can learn the feature extractor automatically through backpropagation. There are usually a large number of convolution layers and activation layers, which provide a better nonlinear fitting capability [15]. Initially, CNN aimed at the classification of entire image and achieved remarkable improvements in ImageNet large scale visual recognition challenge (ILSVRC) [16,17]. After that, some outstanding networks such as visual geometry group (VGG) [18], ResNet [19], and DensNet [20] deepened the network structure and enhanced the accuracy of classification further. However, the fully connected layers at the end of the network destroy the spatial structure of the feature maps, which make it impossible to apply CNN directly into dense semantic labeling tasks [21]. To preserve spatial information, Long [22] proposed the fully convolutional network (FCN), which utilizes upsampling operations for the replacement of fully connected layers. As a result, the extracted low-resolution feature maps can be recovered to the input resolution. FCN is the earliest dense semantic labeling network, and all the later networks follow this idea [23–26].

Currently, the accuracy of dense semantic labeling has been largely improved by the relative models. Classification of small objects and acquiring a sharper object boundary become the main challenge [27]. To this end, deeper backbone networks [28] and the network structure such as encoder-decoder [29] are widely utilized. Deeper backbone networks with more convolution layers can better extract context information, while the encoder-decoder structure can recover the lost spatial information caused by downsampling operations by involving the low-level features from the shallow layers. These methods significantly improve the performance of dense semantic labeling. However, they also come with more network parameters and high computational overheads, which may cause slower inference speed, more hardware consumption and impeding practical large-scale applications.

To maintain efficient inference speed, a large number of works have focused on improving the real-time performance of dense semantic labeling. There are primarily four different approaches to speed up the network. First, the work proposed by Wu [30] downsamples the input image to reduce computational complexity. However, the loss of spatial information in the input leads to an inaccurate prediction around object boundaries. Second, RefineNet [31] compresses the channels of the network especially in the shallow layers to improve the inference speed. However, the simplifying of shallow layers weakens the extraction of spatial information. Third, ENet [32] abandons the last stage of downsampling operation in pursuit of an extreme light-weight structure. Losing the downsampling operation has an obvious shortcoming: The receptive field is insufficient to cover large objects, leading to poor extraction of context information. Lastly, the dual-path structure proposed by BiSeNet [33] introduces two separate sub-networks to extract context information and spatial information respectively. Among the above solutions, the dual-path structure is state-of-the-art, and both spatial and context information can be well extracted.

The dual-path structure consists of two parts: the spatial path and the context path. The spatial path usually adopts a simple structure with several convolution layers to extract high-resolution spatial information, while the context path uses a light-weight backbone network to extract context information. After the feature extraction, it can fuse both kinds of features to get the joint spatial-context features for the final prediction. Due to the respective extraction process, the dual-path structure could overcome the network redundancy and achieve high accuracy without involving much computational overhead.

To improve performance with dual-path structure, the light-weight backbone network in the context path plays an essential role, and a few attempts have been made. MobileNet [34,35] takes the advantages from the depthwise separable convolution. Unlike the standard convolution, depthwise convolution applies a single filter to each input channel, leading to an extreme decrease in computation cost. Afterward, ShuffleNet [36,37] introduces group convolution and channel shuffle, the information

exchange between channels is taken into account with low computational complexity. However, the performance of these light-weight backbone networks is not comparable to the state-of-the-art.

In addition to the network structure and backbone, other components such as channel attention block and pyramid pooling module (PPM) are also helpful in the pursuit of better accuracy without loss of speed. The channel attention block from SENet [38] uses global average pooling followed by fully connected layers and the sigmoid operation to weight each channel according to its importance, and it is an effective filter after feature extraction. The PPM module from PSPNet [39] consists of several branches of pooling operations with different sizes of kernels to enhance the context information after the backbone network. These network components can significantly improve performance without involving too many parameters and computational overheads.

Inspired by the analysis above, in this paper, we propose a novel light-weight CNN to bridge the gap between model efficiency and accuracy for dense semantic labeling of high-resolution remote sensing image. Our model is designed under the efficient dual-path architecture, which can separate the feature extraction process and avoid network redundancy. In the spatial path, a simple structure of three convolution layers with stride two are utilized to preserve affluent spatial information. For the context path, a deep network structure is needed to supply a sufficient receptive field, and we adopt the multi-fiber network (MFNet) [40] as the backbone due to the characteristics of excellent feature extraction ability and low computational complexity.

To further enhance the context features, we analyze the existing methods and append the pyramid pooling module (PPM) after the backbone, without involving too many computational overheads. On top of these two paths, we explore the attention mechanism and apply the channel attention block to refine the context features and the fused spatial-context features further. Moreover, a weighted cascade loss function is developed to guide the training procedure. Unlike the single loss function, the proposed loss function can better optimize the network parameters at different stages. Experiments on the Potsdam and Vaihingen datasets [41,42] demonstrate that the proposed network performs better than other light-weight networks even some state-of-the-art networks and achieves 87.5% and 86.1% overall accuracy respectively with only 8.7 M network parameters and 7.4 G floating point operations (FLOPs). The main contributions of this work are listed as follows:

1. We propose a novel light-weight CNN with a dual-path architecture, which applies MFNet as the backbone network of the context path.
2. We improve the performance by applying the channel attention block and the pyramid pooling module without involving too many computational overheads.
3. We enhance the training procedure by developing a weighted cascade loss function.

The remainder of this paper is organized as follows: Section 2 describes the pre-processing methods, the proposed light-weight network, the datasets and the training protocol. Section 3 presents the metrics and the results. Section 4 is the discussion and Section 5 concludes the whole work.

2. Methodology

In this work, a light-weight dense semantic labeling network is proposed for the high-resolution remote sensing image. The network consists of the following parts. First, we slice the high-resolution images and corresponding ground-truth from the International Society for Photogrammetry and Remote Sensing (ISPRS) 2D contest [41,42] into small patches as the training and test samples. Then, three data augmentation methods are applied to enhance the input samples. Afterward, the proposed light-weight model with the dual-path framework is trained based on the updated parameters, which are calculated by the weighted cascade loss function. Finally, the trained model produces the predictions of the test samples, and the predictions of small patches are spliced to output the final results. The pipeline of the light-weight dense semantic labeling network is shown in Figure 1.

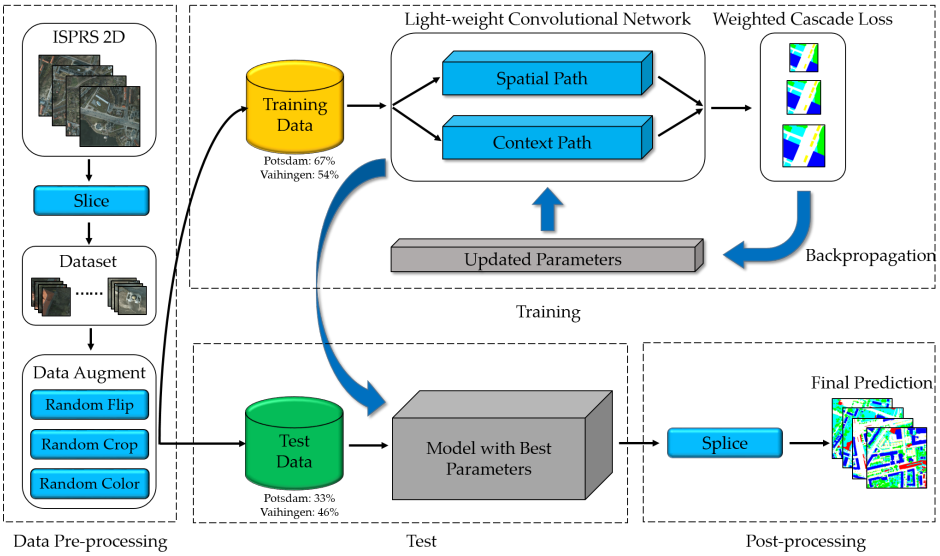


Figure 1. Pipeline of the proposed light-weight dense semantic labeling system, including data pre-processing, training, testing, and post-processing. There were 67% (54%) and 33% (46%) of the sliced patches for training and testing on Potsdam (Vaihingen) dataset respectively. International Society for Photogrammetry and Remote Sensing (ISPRS).

2.1. Pre-Processing Methods

Pre-processing is significant in deep learning based methods. It can enhance the data complexity to achieve better performance. In this section, we introduce the pre-processing methods that we applied to the proposed network, which contain two stages: image slicing and data augmentation.

We evaluate the proposed network on the ISPRS 2D semantic labeling contest, which includes the Potsdam and the Vaihingen datasets. There are 38 images in the Potsdam dataset: 24 images are offered for training and 14 images are preserved for testing. The Vaihingen dataset contains 33 images: 16 images are offered for training, and 17 images are preserved for testing. More details about the two datasets are presented in Section 2.7.1. All the images of the two datasets have a very high resolution, the limitation of GPU memory makes it impossible to put the entire image into the CNN model. The common method to solve this problem is image slicing. In this work, the training images are sliced into small patches with the resolution of 512 pixels $\times$ 512 pixels as the network input. Moreover, an overlay of 64 pixels is also applied to avoid the influence of the slicing boundary. The test images are sliced into the same size without an overlay. As a result, 67% (54%) and 33% (46%) sliced patches are for training and test on the Potsdam (Vaihingen) datasets respectively.

To further improve performance with a given number of samples, data augmentation is an effective way [43]. In this work, three specific data augmentation methods are adopted in the experiments. First, we randomly flip the input patch horizontally and vertically every iteration to deal with the problem of object rotation, which is a common difference between remote sensing images and natural images. Second, objects of the same category may have different sizes, in order to handle this, we randomly resize the input patch with a factor from 0.5–2 every iteration and pad or crop it to the original resolution. Third, seasonal variation and climate change may cause color imbalance problems, resulting in different colors of the same object. To enhance the adaptability of the network to different colors, we randomly adjust the brightness ($-$0.125–0.125), saturation (0.5–1.5), hue ($-$0.2–0.2) and contrast (0.5–1.5) of the input patch every iteration.

2.2. Spatial Path

In this section, we introduce the spatial path of the proposed light-weight dense semantic labeling network. In dense semantic labeling, it has been proven that spatial information and context

information are both crucial to achieve high labeling accuracy [44]. The key idea to better extract spatial information is preserving the resolution of the feature maps, while the extraction of context information needs a deeper network structure and enough receptive field. However, meeting these two requirements simultaneously on a single path may make the network bloat and affect the real-time performance. Therefore, we adopt an exclusive path for the extraction of spatial information.

The objective of the spatial path is to encode enough spatial information for the final prediction. Due to the independent extraction process, it is unnecessary to care about the depth and the receptive field. Thus, we adopt a simple structure of three convolution layers, which extracts spatial information directly from the input image. Each layer comes with stride two followed by batch normalization and rectified linear unit (ReLU) activation function. After that, rich spatial information can be encoded from the high-resolution feature maps, and the output size is 1/8 of the input image. The details of the structure are presented in Figure 2.

2.3. Context Path

In this section, we present the context path of the proposed network. In addition to spatial information, dense semantic labeling also requires sufficient context information to generate a high-quality result. Former works indicate that the extraction of context information relies on the deep network structures. A large number of stacked convolution layers and pooling layers can effectively enlarge the receptive field, while the spatial path has only a shallow network structure. Therefore, an independent context path is needed. Due to the affluent extraction of spatial information by the spatial path, the context path need not consider the complexity of the shallow layers, a light-weight backbone network with fast downsampling and fewer feature map channels can satisfy the requirement.

In this work, we design the context path by introducing the powerful multi-fiber network (MFNet) as the backbone. This network has the characteristics of high performance and efficient computation simultaneously. The pyramid pooling module (PPM) follows to enhance the extraction of context information. Finally, global average pooling is added on the tail to maximize the receptive field. The details of the context path are shown in Figure 2.

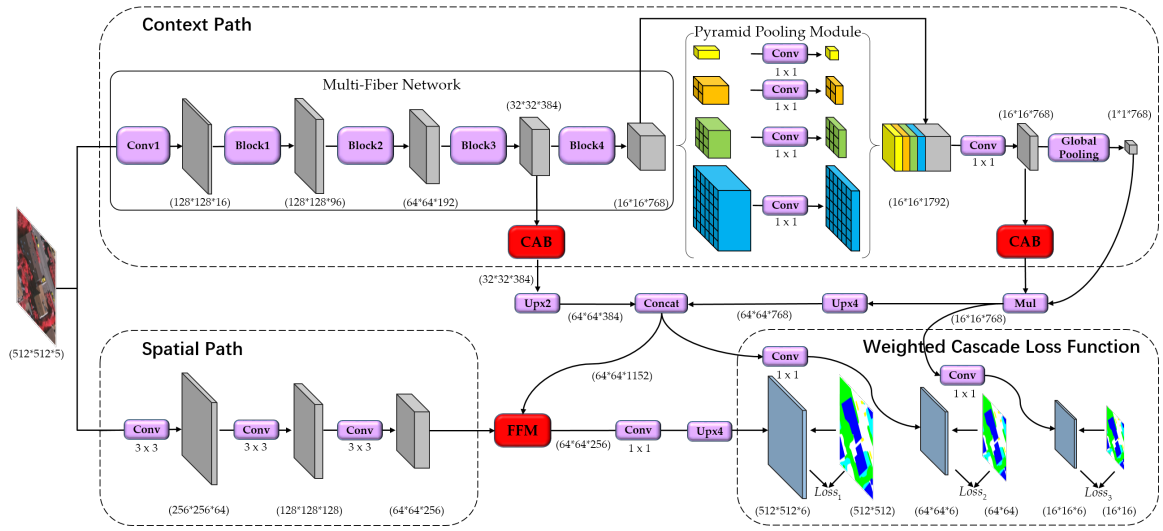


Figure 2. Architecture of the proposed real-time dense semantic labeling network with the dual-path structure. It consists of the spatial path, the context path (multi-fiber network followed by pyramid pooling module), the channel attention block (CAB), the feature fusion module (FFM) and the weighted cascade loss function. Mul: multiplication.

2.3.1. Multi-Fiber Network as the Backbone

The backbone network in the context path is the core structure to extract context information. For real-time dense semantic labeling tasks, a light-weight structure with a high performance of feature extraction is needed. Therefore, we choose the efficient MFNet [40] as the backbone, which was first designed for the video recognition tasks with 3D convolution. In order to apply it to our network, we change all the convolution operations from 3D to 2D. MFNet consists of one shallow convolution layer followed by four blocks, and each block contains 3/4/6/3 units respectively. Similar to the well-known ResNet, MFNet is also based on the specific convolution unit, and the proposed multi-fiber unit plays a key role.

As shown in Figure 3, the channels of the input feature map are divided into N groups, and each group is called one fiber. First, to facilitate the information exchange between different fibers, all the fibers are processed by the multiplexer, which includes two 1×1 convolution layers. After that, each fiber uses two 3×3 convolution layers to extract features. Additionally, the residual shortcut is also adopted in each fiber. Due to the channel grouping and information exchange, MFNet takes both performance and efficiency into account. Specifically, we set the number of fibers to 16 in the proposed model.

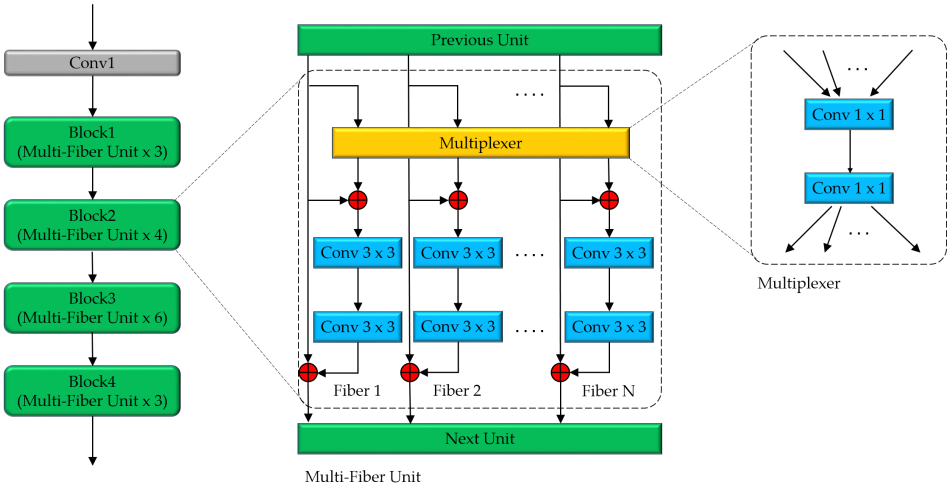


Figure 3. Structure of MFNet, which consists of one convolution layer and four blocks. Each block has several multi-fiber units. Inside the unit, the channels of the input feature map are divided into N fibers, and the multiplexer and residual shortcut are included. In this work, the number of fibers is set to 16.

2.3.2. Pyramid Pooling Module

Appending additional components after the backbone network is an effective way to enhance the extraction of context information further, and the core idea for achieving this objective is to provide the multi-scale receptive field with parallel structures. Currently, a vast number of works have been proposed. Among them, the atrous spatial pyramid pooling (ASPP) from DeepLab [45–47] and the pyramid pooling module (PPM) from PSPNet are the state-of-the-art. Although these two network components look similar, their inner methods are different. ASPP uses atrous convolution with different rates to extract multi-scale features, and it has been proven that atrous convolution is a powerful tool to enlarge the receptive field with holes added in the convolution kernel. PPM is based on the average pooling with different kernel size and stride. As for the real-time dense semantic labeling system, network parameters and computational overheads should be considered, and the pooling operation is more computational friendly than the atrous convolution operation. Therefore, we apply PPM in this work.

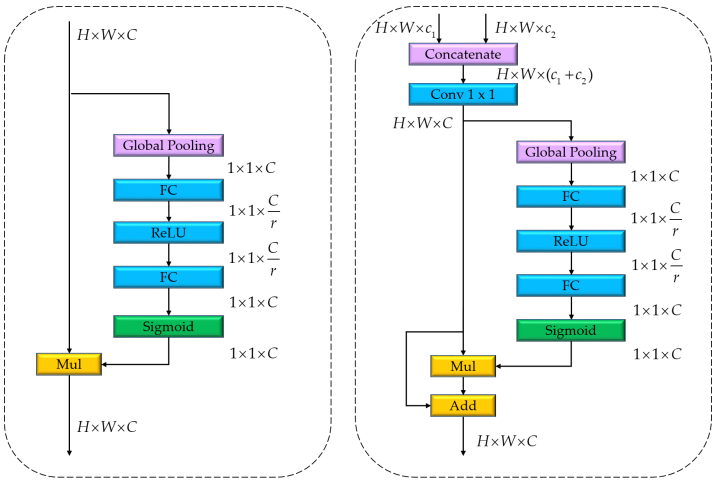
The PPM consists of four parallel branches. The first branch adopts global average pooling to generate a single bin output. The following three branches separate the input feature map into 2×2, 3×3, and 6×6 sub-regions and perform average pooling operation for different locations. Then, a 1×1 convolution is followed in each branch to compress the channel. After that, the output feature maps of each branch are upsampled to get the same resolution as the input via bilinear interpolation, and all the features are concatenated as the output of PPM.

After the PPM, we add a global average pooling layer to maximize the extraction of context information and generate a network tail.

2.4. Channel Attention Block and Feature Fusion Module

In addition to the dual-path structure, we also explore other methods to improve labeling performance with the consideration of computational complexity. Recently, a large number of works focused on the research of attention mechanisms. The attention methods weight the information according to importance, and work like a filter. Among them, the channel attention block proposed by SENet is suitable for real-time dense semantic labeling tasks. As shown in Figure 4a, the input features are first processed by the global average pooling to calculate an attention vector for guiding the feature learning. Then, two fully connected layers and one sigmoid layer are followed to generate the weight for each channel. Finally, the guiding weight is multiplied to the input features. Due to the simple structure, it demands a negligible computation cost. In this work, we adopt the channel attention block to refine two scales of features from the context path.

To fuse spatial information and context information, we adopt the feature fusion module, which also contains the channel attention block. The extracted features of the spatial path and the context path are concatenated, and a 1×1 convolution layer is appended to compress the channel. After that, the channel attention block is followed. In the end, there is a residual shortcut for feature reuse. Figure 4b shows the detailed structure.



(**a**) Channel Attention Block (**b**) Feature Fusion Module

Figure 4. Structures of the channel attention block and the feature fusion module. Mul: multiplication. FC: fully connected layer. ReLU: rectified linear unit.

2.5. Overall Network Architecture

On top of the dual-path structure, we present the overall network architecture of the proposed real-time dense semantic labeling model. We utilize two scales of feature maps from the context path for fusion with the spatial features. The first one is the last feature map before the global average pooling, which is multiplied by the network tail to refine the context information, while the second one comes from the output of the intermediate layers in the backbone network (Block3 of MFNet). Both features are processed by the channel attention module. Then, these two features are upsampled

4/2 times respectively to meet the same resolution, and concatenated as the overall context information. Finally, the spatial features and the context features are fused by the feature fusion module to output the final prediction. Details of the overall network architecture are shown in Figure 2. During the training procedure, we develop a weighted cascade loss function for the proposed model.

2.6. Cascade Loss Function

The loss function is a necessary network component in the deep learning model to calculate the deviation value between the prediction and the ground-truth through backpropagation. In dense semantic labeling tasks, the softmax cross-entropy loss function is the commonly used one. However, the pixel number of each categorized ground object varies significantly in the remote sensing image, and the original cross-entropy loss is heavily affected by the class distribution imbalance. To handle this problem, we apply the median frequency balance strategy [48], which gives a class weight for each category. The expression of weighted cross-entropy loss is:

$$Loss = -\frac{1}{N}\sum_{i=1}^{N} W_i \cdot \tilde{p}_i \cdot log\left(\frac{e^{p_i}}{\sum_{j=1}^{N} e^{p_j}}\right) \tag{1}$$

where N is the number of categories, W_i denotes the weight for category i, p_i and $\tilde{p}_i$ denotes the prediction and the ground-truth distribution of category i respectively.

Most of the traditional dense semantic labeling models apply one loss function at the end of the network. For our model, we adopt the dual-path structure, which consists of three main parts: the spatial path, the context path, and the feature fusion module. This is a more complex situation for the backpropagation and one loss function is insufficient to optimize all the layers especially the layers in the context path (far from the end of the network). The final prediction depends on three scales of feature maps: the spatial features, and two scales of context features. Therefore, we develop a weighted cascade loss function, which consists of three losses to better supervise the training procedure. First, we append a loss function at the end of the network with the ground-truth of the original resolution. Then, another loss function is applied after the concatenation of the two scales context information with the eight times downsampled ground-truth. Finally, the last loss function is employed on the feature map at the end of the context path with the 32 times downsampled ground-truth. The first loss function at the end guides the training of the whole network as most methods, while the two latter ones in the middle can further enhance the optimization of the parameters in the context path. To emphasize their importance, we also put weights for these three losses as λ_1, λ_2, and λ_3. The expression of the weighted cascade loss function is:

$$Loss_{overall} = \lambda_1 \times Loss_1 + \lambda_2 \times Loss_2 + \lambda_3 \times Loss_3. \tag{2}$$

Section 4.4 shows the analysis of the proposed weighted cascade loss function. The details are shown in Figure 2.

2.7. Experimental Settings

2.7.1. Datasets

The proposed real-time dense semantic labeling network is evaluated on the Potsdam and the Vaihingen datasets, which are provided by the ISPRS 2D semantic labeling contest and are open access to the public [42]. All the images are true high-resolution orthophotos and collected in urban scenes by airborne sensors. There are six labeled categories in both datasets: impervious surfaces, building, low vegetation, tree, car, and clutter/background. The clutter/background class includes the objects that are not of interest in the dense semantic labeling in urban scenes (e.g., tennis courts, swimming pools, and containers). Figure 5 shows the study area of these two datasets.

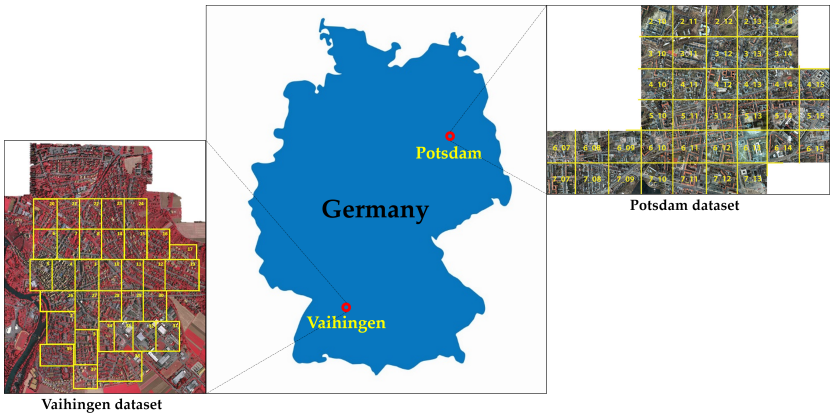


Figure 5. Study areas of the Potsdam dataset and the Vaihingen dataset [41,42].

The Potsdam dataset shows the scene of the historic city Potsdam in Germany. It consists of 38 images with the resolution of 6000 pixels × 6000 pixels. Among them, 24 images are offered for training and 14 images are preserved for testing. Each image contains 4-channel data of red, green, blue, and near infra-red (IR) with the corresponding digital surface model (DSM) generated via dense image matching. The spatial resolution of each image and the corresponding DSM is about 0.05 m, and the manually annotated ground-truth of pixel-level is available for all the images. In this work, we concatenate the 4-channel image and the normalized DSM (nDSM) to form 5-channel data as the network input. Therefore, the input channel of the first convolution layers in both paths is set to 5 in the experiments related to the Potsdam dataset.

The Vaihingen dataset shows the scene of a relatively small village, Vaihingen, in Germany. It consists of 33 images with the approximate resolution of 2500 pixels × 2500 pixels. Among them, 16 images are offered for training and 17 images are preserved for testing. Each image contains 3-channel data of red, green, and near infra-red with the corresponding DSM. The spatial resolution of each image and the corresponding DSM is about 0.09 m, and the manually annotated ground-truth of pixel-level is available for all the images too. In this work, we concatenate the 3-channel image and the nDSM to form a 4-channel data as the network input. Therefore, the input channel of the first convolution layers in both paths is set to 4 in the experiments related to the Vaihingen dataset. The detailed descriptions are shown in Table 1, and Figures 6 and 7 shows example images of these two datasets respectively.

Table 1. Detailed descriptions of the Podsdam and the Vaihingen datasets.

Dataset	Resolution (pixel)	Spatial Resolution	Area	Bands	DSM	Angle
Potsdam	6000 × 6000	0.05 m	city	R, G, B, IR	√	otho
Vaihingen	≈2500 × 2500	0.09 m	village	R, G, IR	√	otho

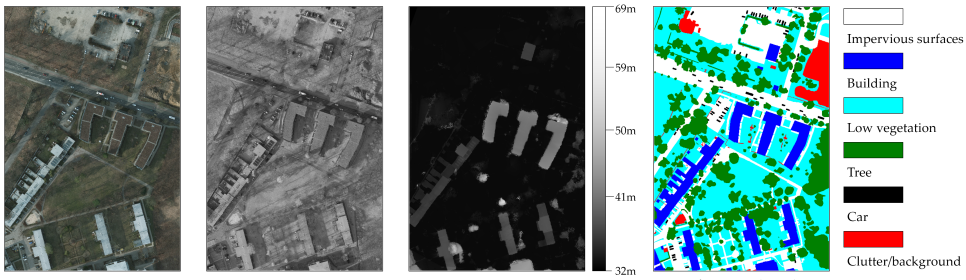
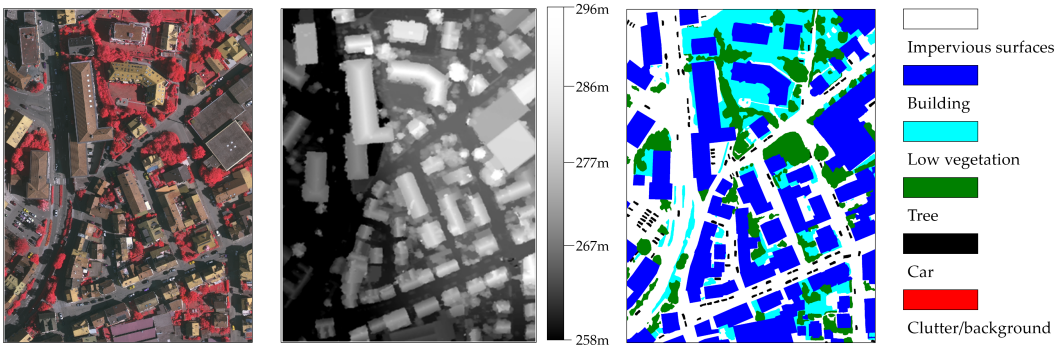


Figure 6. Samples of remote sensing images in the Potsdam dataset. (**a**) RGB image. (**b**) Near infra-red (IR). (**c**) Normalized digital surface model (DSM). (**d**) Manually annotated ground-truth.



(**a**) RGIR (**b**) nDSM (**c**) Ground-truth

Figure 7. Samples of remote sensing images in the Vaihingen dataset. (**a**) RGIR image. (**b**) Normalized DSM. (**c**) Manually annotated ground-truth.

2.7.2. Training Protocol

The proposed real-time dense semantic labeling model is deployed on the PyTorch platform. The hardware configurations are i7-4790k, 16 GB RAM with one NVIDIA GTX1080TI GPU (11 GB RAM). We set the batch size to six due to the memory limitation of the single GPU and adopted the adaptive moment estimation (ADAM) [49] as the optimizer. For the learning rate, we utilize the poly policy, the expression is:

$$lr = initial\_lr \left(1 - \frac{iteration}{max\_iteration}\right)^{power} \tag{3}$$

where $initial\_lr = 0.01$, $power = 0.9$, and $max\_iteration = 100,000$ in this work. For the weights of the weighted cascade loss function, when $\lambda_1 = 1$, $\lambda_2 = 0.4$, and $\lambda_3 = 0.16$, the best performance could be achieved on both datasets. During the training procedure, we perform validation using all the test data every 2000 iterations.

3. Results

3.1. Metrics

In this work, four different metrics are employed to evaluate the performance of the proposed model: overall accuracy (OA), F_1 score, precision, and recall. All of them are widely involved in the former works [41,50]. The expressions of them are shown as follows:

$$OA = \frac{1}{n}\sum_{i=1}^{n} W_i \cdot \frac{TP_i + TN_i}{P_i + N_i} \tag{4}$$

$$Precision = \frac{1}{n}\sum_{i=1}^{n} W_i \cdot \frac{TP_i}{TP_i + FP_i} \tag{5}$$

$$Recall = \frac{1}{n}\sum_{i=1}^{n} W_i \cdot \frac{TP_i}{TP_i + FN_i} \tag{6}$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{7}$$

where n is the number of categories, W_i denotes the sample weight of category i, N_i is the number of negative samples of category i, P_i is the number of positive samples of category i, TP_i represents the true positive of category i, FP_i represents the false positive of category i, and FN_i is the false negative of category i.

The proposed network focuses on real-time dense semantic labeling, along with the above metrics, we also adopt the number of network parameters, floating point operations (FLOPs) and inference time

as additional metrics for the evaluation of real-time performance. The number of network parameters and FLOPs are calculated using the open source PyTorch-OpCounter (THOP) package.

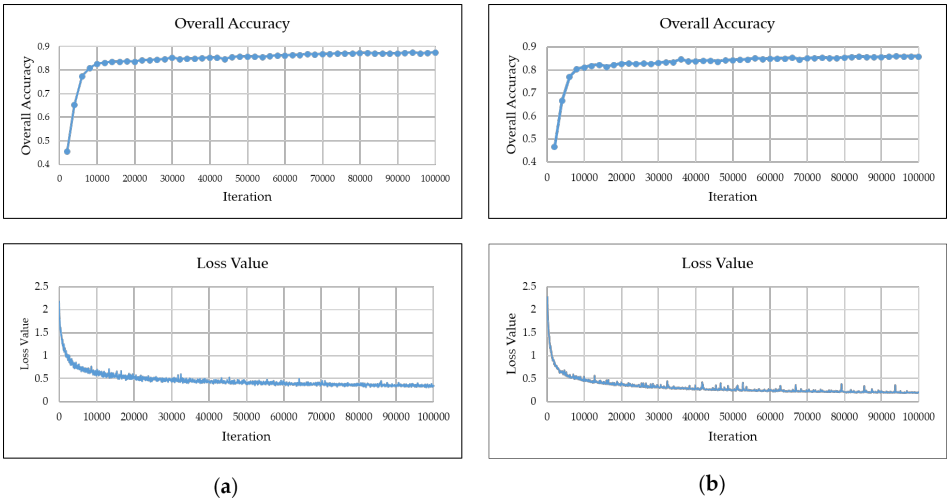
3.2. Results of the Proposed Model

To better evaluate the proposed real-time dense semantic labeling network, classic networks such as FCN, U-Net [51], DeepLab_v3 are utilized as the baseline for the comparison to our model. Moreover, some well-known real-time networks such as ENet, ICNet [52], and BiSeNet are also included in the comparison. All 14 test images in the Potsdam dataset and 17 test images in the Vaihingen dataset are used for evaluation. It should be noted that all the metric scores are computed with the pixels of the object boundary.

The proposed model achieves 87.5% overall accuracy on the Potsdam dataset and 86.1% overall accuracy on the Vaihingen dataset. For real-time performance, our model only has 8.7 M parameters and 7.4 G FLOPs with the input size of 512 pixels × 512 pixels. The training procedure of the 100,000 iterations takes about 6 h. Figure 8 shows the training plot, the convergence process is stable, and the overall accuracy exceeds 80% with a fast speed on two datasets. The sample results of the proposed model on the Potsdam and the Vaihingen datasets are shown in Figures 9 and 10 respectively. The first column shows the high-resolution remote sensing images; the second column is the corresponding ground-truth; and the last column represents the prediction results. Detailed metric scores of accuracy on the two datasets are shown in Table 2. From the results, we can see that our model successfully reaches the balance of high accuracy and efficiency.

Table 2. Metric scores of overall accuracy, precision, recall, and F_1 on the Potsdam and the Vaihingen datasets.

Dataset	Metrics	Imp_surf	Building	Low_veg	Tree	Car	Average
Potsdam	OA	N/A	N/A	N/A	N/A	N/A	0.875
	Precision	0.886	0.949	0.804	0.861	0.880	0.876
	Recall	0.914	0.965	0.861	0.798	0.881	0.884
	F_1	0.899	0.957	0.831	0.825	0.880	0.878
Vaihingen	OA	N/A	N/A	N/A	N/A	N/A	0.861
	Precision	0.874	0.910	0.785	0.835	0.783	0.837
	Recall	0.885	0.923	0.762	0.871	0.714	0.831
	F_1	0.879	0.916	0.773	0.853	0.747	0.834



(**a**) (**b**)

Figure 8. Training plot of loss value and overall accuracy on two datasets. (**a**) The loss value curve and overall accuracy curve on Potsdam dataset. (**b**) The loss value curve and overall accuracy curve on Vaihingen dataset.

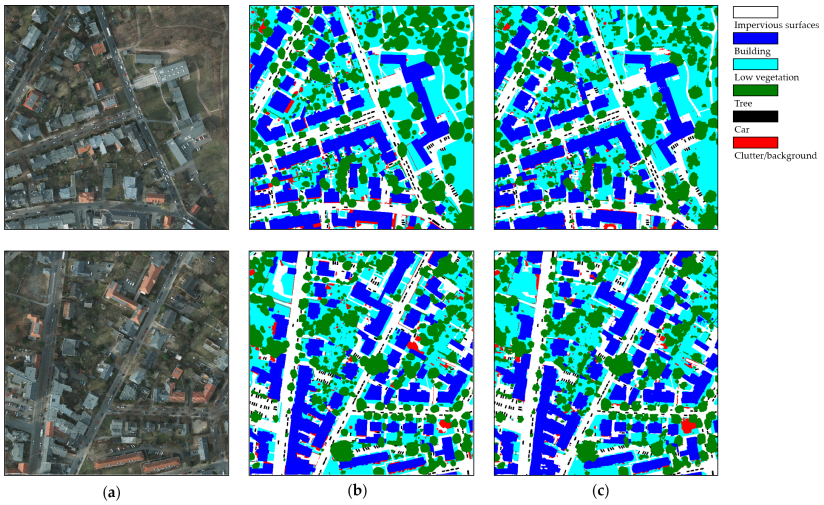


Figure 9. Example results of the proposed model on the Potsdam dataset. (**a**) RGB remote sensing images. (**b**) Manually annotated ground-truth. (**c**) Prediction maps of the proposed model.

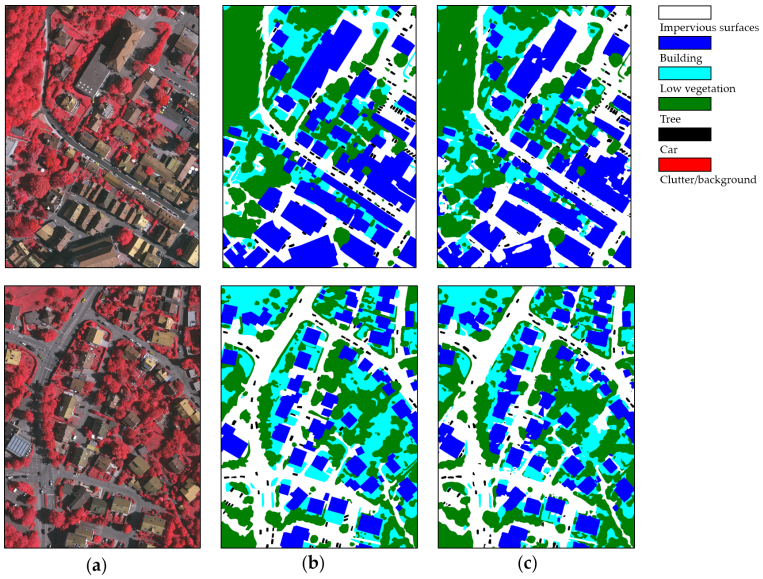


Figure 10. Example results of the proposed model on the Vaihingen dataset. (**a**) RGIR remote sensing images. (**b**) Manually annotated ground-truth. (**c**) Prediction maps of the proposed model.

3.3. Comparison with the State-of-the-Art

In this section, we evaluate the proposed model in comparison with other real-time dense semantic labeling models, even some classic or state-of-the-art models. The real-time models include ENet, ICNet, and BiSeNet. ENet is the earliest real-time dense semantic labeling model, which has been widely used in embedded devices because of the extremely fast speed. ICNet proposes a cascade network based on PSPNet and achieves a balance between accuracy and speed. BiSeNet is a newly published network, which proposes the dual-path architecture and improves accuracy and speed to a higher level. Classic or state-of-the-art models include FCN, U-Net, and DeepLab_v3. FCN is the first end-to-end dense semantic labeling model, which can still be a baseline. U-Net is the most important network in the research of remote sensing images, which is also key in comparison to the proposed model. DeepLab_v3 is a state-of-the-art model and represents the highest level of dense semantic labeling. As shown in Tables 3 and 4, the proposed model has the best labeling accuracy among the real-time models, and the network parameters and FLOPs are lower than ICNet and BiSeNet. Compared to the classic or the state-of-the-art networks, our model outperforms U-Net on the two

datasets and has 2.5 times less network parameters and 22 times less computational FLOPs. While the accuracy gap with DeepLab_v3 is small, and the speed has been largely improved. The training procedure of DeepLab_v3 of the 100,000 iterations takes about 20 h. Figures 11 and 12 show examples of the comparison results.

Table 3. Real-time comparison results of different networks including ENet, ICNet, BiSeNet, FCN, U-Net, and DeepLab_v3 on the Potsdam and the Vaihingen datasets. The input size is 512 pixels × 512 pixels. Floating point operations (FLOPs), overall accuracy (OA).

Method	FLOPs	Parameters	Inference Time	OA (Potsdam)	OA (Vaihingen)
ENet [32]	4.2 G	0.4 M	6 ms	0.783	0.772
ICNet [50]	40.4 G	26.5 M	18 ms	0.837	0.824
BiSeNet [33]	12.3 G	12.1 M	10 ms	0.865	0.853
FCN [22]	124.4 G	184.9 M	41 ms	0.824	0.810
U-Net [49]	165.9 G	22.0 M	59 ms	0.860	0.848
DeepLab_v3 [44]	241.7 G	58.1 M	106 ms	0.878	0.863
Ours	7.4 G	8.7 M	8 ms	0.875	0.861

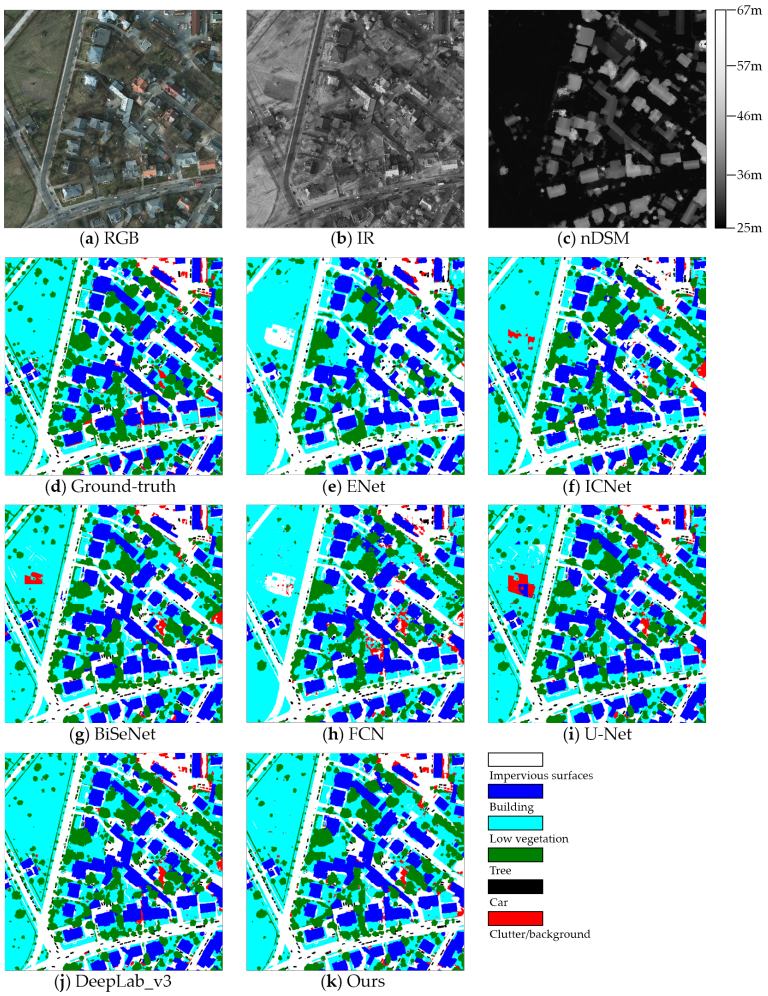


Figure 11. A sample of the comparison prediction results from different models on the Potsdam dataset. (**a**) RGB image. (**b**) Near infra-red. (**c**) normalized DSM. (**d**) Corresponding ground-truth. (**e**) Result of ENet. (**f**) Result of ICNet. (**g**) result of BiSeNet. (**h**) Result of FCN. (**i**) Result of U-Net. (**j**) Result of DeepLab_v3. (**k**) Result of our model.

Table 4. Quantitative results (F_1, OA) of different networks including ENet, ICNet, BiSeNet, FCN, U-Net, and DeepLab_v3 on the Potsdam and the Vaihingen datasets.

Dataset	Methods	Imp_surf	Building	Low_veg	Tree	Car	Average F_1	Overall Accuracy
Potsdam	ENet [32]	0.830	0.915	0.718	0.637	0.568	0.733	0.783
	ICNet [52]	0.865	0.943	0.771	0.768	0.810	0.832	0.837
	BiSeNet [33]	0.884	0.951	0.816	0.819	0.867	0.867	0.865
	FCN [22]	0.856	0.933	0.770	0.741	0.757	0.812	0.824
	U-Net [51]	0.884	0.934	0.808	0.825	0.878	0.866	0.860
	Deeplab_v3 [46]	0.900	0.958	0.831	0.835	0.885	0.882	0.878
	Ours	0.899	0.957	0.831	0.825	0.880	0.878	0.875
Vaihingen	ENet [32]	0.815	0.876	0.678	0.622	0.560	0.710	0.772
	ICNet [52]	0.840	0.906	0.693	0.772	0.695	0.781	0.824
	BiSeNet [33]	0.878	0.915	0.762	0.842	0.733	0.826	0.853
	FCN [22]	0.830	0.877	0.690	0.826	0.620	0.769	0.810
	U-Net [51]	0.875	0.895	0.751	0.852	0.744	0.823	0.848
	Deeplab_v3 [46]	0.881	0.918	0.774	0.861	0.751	0.837	0.863
	Ours	0.879	0.916	0.773	0.853	0.747	0.834	0.861

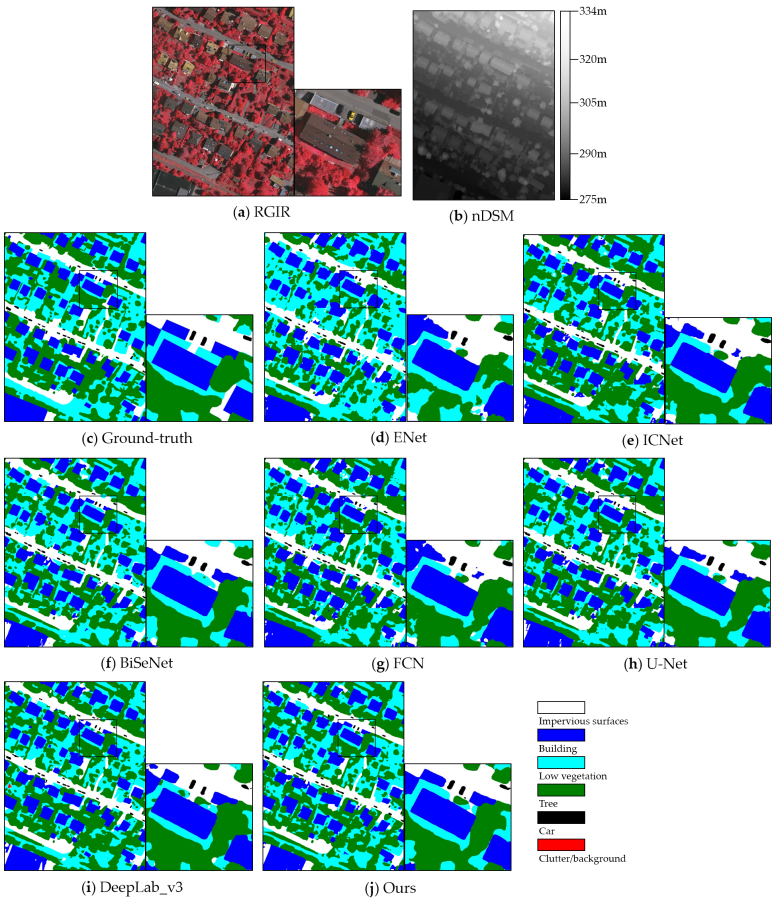


Figure 12. A sample of the comparison prediction results from different models on the Vaihingen dataset. (**a**) RGIR image. (**b**) Normalized DSM. (**c**) Corresponding ground-truth. (**d**) Result of ENet. (**e**) Result of ICNet. (**f**) Result of BiSeNet. (**g**) Result of FCN. (**h**) Result of U-Net. (**i**) Result of DeepLab_v3. (**j**) Result of our model.

4. Discussions

4.1. Effects of the Backbone Network

The backbone network in the context path is the most important part of feature extraction, which has a significant influence on the final labeling accuracy. Meanwhile, most of the convolution layers, network parameters, and computational overheads of the whole network are concentrated in it. Therefore, both the performance and the efficiency of the backbone network should be focused on consideration in the real-time dense semantic labeling model. In this work, we adopt the MFNet as the backbone in the context path. To better evaluate it, we also did comparative experiments with other light-weight networks, including ResNet-18 and MobileNet-v2. ResNet-18 is the smallest version of ResNet, and MobileNet-v2 is a well-known backbone based on depthwise separable convolution, both of them are widely used in the former works. As shown in Table 5, the model with MFNet has the fewest network parameters and achieves the best labeling accuracy on both datasets. The FLOPs and the inference time is just a little higher than the model with MobileNet-v2. The experimental results indicate that MFNet is the better choice to meet the balance between accuracy and efficiency.

Table 5. Comparison results of the proposed model with a different backbone in the context path on the Potsdam (P) and the Vaihingen (V) datasets. The input size is 512 pixels × 512 pixels.

Backbone	FLOPs	Parameters	Inference Time	F_1 (P)	OA (P)	F_1 (V)	OA (V)
ResNet-18	12.5 G	14.6 M	11 ms	0.871	0.869	0.828	0.854
MobileNet-v2	6.6 G	9.8 M	7 ms	0.868	0.864	0.825	0.852
Ours (MFNet)	7.4 G	8.7 M	8 ms	0.878	0.875	0.834	0.861

4.2. Influence of the Pyramid Pooling Module

It has been proven that appending network components with parallel structures after the backbone network is an effective way to enhance the extraction of context information further. The ASPP from DeepLab and the PPM from PSPNet are the state-of-the-art methods. Both of them can enlarge the receptive field at different scales. In the context path, we apply PPM after MFNet due to the efficiency of the pooling operation compared to the atrous convolution operation. To support this statement, we did experiments with these two components respectively and deploy a sub-network without any of them as the baseline. As shown in Table 6, the model with PPM has an obvious improvement to the baseline. The overall accuracy is increased by about 0.5% on two datasets with slightly higher FLOPs and network parameters. While the accuracy of the model with ASPP is improved less, the FLOPs and network parameters are increased more. Therefore, the PPM network component is more suitable for the real-time dense semantic labeling network.

Table 6. Comparison results of the proposed model with the pyramid pooling module (PPM) or the atrous spatial pyramid pooling (ASPP) after the backbone in the context path on the Potsdam (P) and the Vaihingen (V) datasets. The basline is the model with none of them. The input size is 512 pixels × 512 pixels.

Model	FLOPs	Parameters	Inference Time	F_1 (P)	OA (P)	F_1 (V)	OA (V)
Baseline	7.2 G	8.6 M	8 ms	0.874	0.870	0.829	0.855
ASPP	8.5 G	10.1 M	9 ms	0.877	0.873	0.832	0.860
PPM	7.4 G	8.7 M	8 ms	0.878	0.875	0.834	0.861

4.3. Effects of the Channel Attention Block (CAB)

The channel attention block proposed by SENet is an efficient network component to refine the extracted features, and it adopts a simple structure based on the global average pooling to generate the weights for each channel. For the real-time dense semantic labeling tasks, this simple structure

does not bring too many computational overheads, which means it is a perfect way to improve the labeling performance further. In this work, we utilize the channel attention block to refine the features from the context path at two scales and also employ it in the feature fusion module to refine the fused spatial-context information. To evaluate the performance, we did additional experiments without the channel attention block. The detailed results are shown in Table 7. From the results, we can see that the increase of the FLOPs is too small to be seen with the deployment of the channel attention block, and the number of network parameters is slightly increased. Meanwhile, the overall accuracy has been improved by about 0.3%. Therefore, the application of the channel attention block in the proposed model has a positive effect.

Table 7. Comparison results of the proposed model with or without the channel attention block (CAB) on the Potsdam (P) and the Vaihingen (V) datasets. The input size is 512 pixels × 512 pixels.

Model	FLOPs	Parameters	Inference Time	F_1 (P)	OA (P)	F_1 (V)	OA (V)
Without CAB	7.4 G	8.6 M	8 ms	0.876	0.872	0.832	0.859
With CAB	7.4 G	8.7 M	8 ms	0.878	0.875	0.834	0.861

4.4. Importance of the Weighted Cascade Loss Function

The loss function is a necessary component for the network training in the deep learning models. In this work, the proposed network is designed based on the dual-path architecture, and the final prediction depends on three scale features: the spatial features and the two scales of context features. Due to the complexity of this structure, traditional single loss function at the end of the network is insufficient to optimize all the layers, especially the layers in the context path. Therefore, we develop a weighted cascade loss function at three stages to better guide the training procedure. To test the improvement of the weighted cascade loss function, we did comparison experiments with the single loss function and the cascade loss function with equal weights. As shown in Table 8, the model trained with the weighted cascade loss function achieves the best labeling accuracy. It should be noted that the loss function is only used during the training procedure and has no influence on the real-time performance of the inference phase. Therefore, we only list the labeling accuracy here.

Table 8. Comparison results of the proposed model trained with single loss function or the cascade loss function on the Potsdam and the Vaihingen datasets.

Potsdam	Precision	Recall	F_1	OA
Single Loss	0.871	0.875	0.873	0.868
Cascade Loss (equal weights)	0.875	0.880	0.877	0.872
Weighted Cascade Loss	0.876	0.884	0.878	0.875
Vaihingen	**Precision**	**Recall**	F_1	**OA**
Single Loss	0.830	0.818	0.824	0.852
Cascade Loss (equal weights)	0.835	0.825	0.830	0.857
Weighted Cascade Loss	0.837	0.831	0.834	0.861

4.5. Analysis of Wide-Spread Applicability

In this section, we analyze the wide-spread applicability of the proposed network. The main objective of the proposed light-weight dense semantic labeling network is to improve the real-time performance of dense semantic labeling with a high labeling accuracy. From the results of the Potsdam and the Vaihingen datasets in Tables 3 and 4, we can see that the proposed network achieved this objective in urban areas. The labeling accuracy is competitive, and there are no obvious shortcomings in each category compared to other existing methods, while the inference time and computational overheads have been largely reduced. Therefore, the proposed network is more suitable for practical applications in urban areas.

However, for the rural areas, all the CNN based methods have the same drawback. The labeling accuracy of the categories of the tree and low vegetation is lower to the categories of buildings and impervious surfaces, which may lead to a decrease in labeling performance. The reason for this phenomenon may come from two aspects. First, trees and low vegetation usually have irregular shapes, which makes CNN hard to obtain a sharp boundary. Second, the multi-scale nature of trees makes it difficult for CNN to label all of them, especially the small trees. To further improve the labeling performance of the CNN based methods in rural areas, the extraction of multi-scale objects and retaining more spatial information should be considered in the future works.

For a different dataset with the same type of data, the proposed network may also achieve a high labeling accuracy. The reason may come from two aspects. First, there are no obvious shortcomings in the accuracy of every category of the object compared to other existing networks. Second, the labeling accuracy of the repeated experiment is stable. For a dataset captured by a different imaging sensor, such as WordView III, the proposed network may achieve higher performance with the increase of data channels. More bands of data can supply additional information for the ground objects, which makes it easier for the networks to extract linear separable features for each category.

5. Conclusions

Real-time dense semantic labeling for high-resolution remote sensing images is a challenging task due to the consideration of precision and efficiency simultaneously. In this paper, an efficient light-weight CNN with the dual-path architecture is proposed to handle this issue. Our model utilizes a simple structure of three convolution layers as the spatial path to preserve spatial information. Meanwhile, we develop the context path with the efficient multi-fiber network followed by the pyramid pooling module to obtain a sufficient receptive field. Moreover, the channel attention block is adopted to refine the extracted context features and fused spatial-context features. During the training procedure, we append a weighted cascade loss function at different stages for the optimization of the network parameters. Experiments on the Potsdam and Vaihingen datasets demonstrate that the proposed network achieves a high labeling accuracy with fast speed and small computation cost. Compared to the classic U-Net, our model achieves higher labeling accuracy on the two datasets with 2.5 times less parameters and 22 times less computational FLOPs.

With the rapid development of remote sensing, more high-quality images are collected for practical applications. However, most existing methods are fully supervised based on manually annotated ground-truth, which takes too much labor and becomes the biggest challenge. In the future works, semi-supervised or weakly supervised methods should be considered. Semi-supervised methods using the generative adversarial network (GAN) can extend the dataset by generating new training samples, which may achieve the same labeling performance with fewer training samples. While weakly supervised methods based on image-level or bounding box ground-truth usually adopt traditional machine learning approaches to obtain a coarse pixel-level ground-truth and train the model with it, both kinds of methods can significantly save labor and will be the direction of future research.

Author Contributions: Conceptualization, Y.W., C.C. and M.D.; methodology, Y.W.; software, Y.W.; validation, Y.W.; formal analysis, Y.W.; investigation, Y.W.; resources, J.L.; data curation, Y.W.; writing—original draft preparation, Y.W.; writing—review and editing, C.C., M.D. and J.L.; visualization, Y.W.; supervision, C.C., M.D. and J.L.; project administration, J.L.; funding acquisition, J.L.

Funding: This research was funded by the National Natural Science Foundation of China under Grant 61473034 and the Fundamental Research Funds for the China Central Universities of USTB (FRF-DF-19-002).

Acknowledgments: We thank the ISPRS for providing the Potsdam and Vaihingen datasets.

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

Abbreviations

The following abbreviations are used in this manuscript:

ADAM	Adaptive Moment Estimation
ASPP	Atrous Spatial Pyramid Pooling
CAB	Channel Attention Block
CNN	Convolutional Neural Network
DSM	Digital Surface Model
FCN	Fully Convolutional Network
FFM	Feature Fusion Module
FLOP	Floating Point Operation
GPU	Graphics Processing Unit
ISPRS	International Society for Photogrammetry and Remote Sensing
PPM	Pyramid Pooling Module
VGG	Visual Geometry Group

References

1. Moser, G.; Serpico, S.B.; Benediktsson, J.A. Land-cover mapping by Markov modeling of spatial-contextual information in very-high-resolution remote sensing images. *Proc. IEEE* **2013**, *101*, 631–651. [CrossRef]
2. Li, M.; Zang, S.; Zhang, B.; Li, S.; Wu, C. A review of remote sensing image classification techniques: The role of spatio-contextual information. *Eur. J. Remote Sens.* **2014**, *47*, 389–411. [CrossRef]
3. Xu, Y.; Wu, L.; Xie, Z.; Chen, Z. Building extraction in very high resolution remote sensing image using deep learning and guided filters. *Remote Sens.* **2018**, *10*, 144. [CrossRef]
4. Xin, P.; Jian, Z. High-resolution remote sensing image classification method based on convolutional neural network and restricted conditional random field. *Remote Sens.* **2018**, *10*, 920.
5. Kampffmeyer, M.; Arnt-Borre, S.; Robert, J. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 1–9.
6. Liu, S.; Ding, W.; Liu, C.; Liu, Y.; Wang, Y.; Li H. ERN: Edge loss renforced semantic segmentation network for remote sensing images. *Remote Sens.* **2018**, *10*, 1339. [CrossRef]
7. Guo, S.; Jin, Q.; Wang, H.; Wang, X.; Wang, Y.; Xiang, S. Learnable gated convolutional neural network for semantic segmentation in remote-sensing images. *Remote Sens.* **2019**, *11*, 1922. [CrossRef]
8. Marmanis, D.; Wegner, J.D.; Galliani, S.; Schindler, K.; Datcu, M. Semantic segmentation of aerial images with an ensemble of CNNs. *ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *3*, 473. [CrossRef]
9. Michele, V.; Devis, T. Dense semantic labeling of subdecimeter resolution images with convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 881–893.
10. Wang, H.; Wang, Y.; Zhang, Q.; Xiang, S.; Pan, C. Gated convolutional neural network for semantic segmentation in high-resolution images. *Remote Sens.* **2017**, *9*, 446. [CrossRef]
11. Yansong, L.; Sankaranarayanan, P.; Sildomar, T.M.; Eli, S. Dense semantic labeling of very-high-resolution aerial image and LiDAR with fully-convolutional neural networks and higher-order CRFs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 76–85.
12. Hyeonwoo, N.; Seunghoon, H.; Bohyung, H.; Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE International Conference on Computer Vision, Washington, DC, USA, 3–7 December 2015; pp. 1520–1528.
13. Wu, G.; Shao, X.; Guo, Z.; Chen, Q.; Yuan, W.; Shi, X.; Xu, Y.; Shibasaki, R. Automatic building segmentation of aerial image using multi-constraint fully convolutional networks. *Remote Sens.* **2018**, *10*, 407. [CrossRef]
14. Wang, J.; Shen, L.; Qiao, W.; Dai, Y.; Li, Z. Deep feature fusion with integration of residual connection and attention model for classification of VHR remote sensing images. *Remote Sens.* **2019**, *11*, 1617. [CrossRef]
15. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.

16. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Li, K.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
17. 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, 8–10 June 2015; pp. 1–9.
18. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
19. 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 (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
20. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 4700–4708.
21. Wei, X.; Fu, K.; Gao, X.; Yan, M.; Sun, X.; Chen, K.; Sun, H. Semantic pixel labelling in remote sensing images using a deep convolutional encoder-decoder model. *Remote Sens. Lett.* **2018**, *9*, 199–208. [CrossRef]
22. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
23. Fu, G.; Liu, C.; Zhou, R.; Sun, T.; Zhang, Q. Classification for high resolution remote sensing image using a fully convolutional network. *Remote Sens.* **2017**, *9*, 498. [CrossRef]
24. Cheng, W.; Yang, W, Wang, M.; Wang, G.; Chen, J. Context aggregation network for semantic labeling in aerial images. *Remote Sens.* **2019**, *11*, 1158. [CrossRef]
25. Papadomanolaki, M.; Vakalopoulou, M.; Karantzalos, K. A novel object-based deep learning framework for semantic segmentation of very high-resolution remote sensing data: comparison with convolutional and fully convolutional networks. *Remote Sens.* **2019**, *11*, 684. [CrossRef]
26. Peng, C.; Zhang, X.; Yu, G.; Luo, G.; Sun, J. Large kernel matters-improve semantic segmentation by global convolutional network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1743–1751.
27. Wang, Y.; Liang, B.; Ding, M.; Li, J. Dens semantic labeling with atrous spatial pyramid pooling and decoder for high-resolution remote sensing image. *Remote Sens.* **2019**, *11*, 20. [CrossRef]
28. Szegedy, C.; Loffe, 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; pp. 4278–4284.
29. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for scene segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
30. Wu, Z.; Shen, C.; Hengel, A. Real-time semantic image segmentation via spatial sparsity. *arXiv* **2017**, arXiv:1712.00213.
31. Lin, G.; Milan, A.; Shen, C.; Reid, I.D. RefineNet: Multi-path refinement networks for high-resolution semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 5168–5177.
32. Paszke, A.; Chaurasia, A.; Kim, S.; Culurciello, E. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv* **2016**, arXiv:1606.02147.
33. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 325–341.
34. 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.
35. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 19–21 June 2018; pp. 4510–4520.

36. Zhang, X.; Zhou, X.; Lin, M.; Sun, J. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 19–21 June 2018; pp. 6848–6856.
37. 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.
38. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 19–21 June 2018; pp. 7132–7141.
39. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
40. Chen, Y.; Kalantidis, Y.; Li, J.; Yan, S.; Feng, J. Multi-fiber networks for video recognition. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 352–367.
41. Gerke, M. *Use of the Stair Vision Library within the ISPRS 2D Semantic Labeling Benchmark (Vaihingen)*; Technical Report; University of Twente: Enschede, The Netherlands, 2015.
42. ISPRS 2D Semantic Labeling Contest. Available online: http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html (accessed on 10 December 2019).
43. Liu, Y.; Ren, Q.; Geng, J.; Ding, M.; Li, J. Efficient Patch-Wise Semantic Segmentation for Large-Scale Remote Sensing Images. *Sensors* **2018**, *18*, 3232. [CrossRef] [PubMed]
44. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
45. Chen, L.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef] [PubMed]
46. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
47. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv* **2018**, arXiv:1802.02611.
48. Pan, X.; Gao, L.; Zhang, B.; Yang, F.; Liao, W. High-resolution aerial image semantic labeling with dense pyramid network. *Sensors* **2018**, *18*, 3774. [CrossRef]
49. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
50. Kahaki, S.M.; Arshad, H.; Nordin, M.J.; Ismail, W. Geometric feature descriptor and dissimilarity-based registration of remotely sensed image. *PLoS ONE* **2018**, *13*, e0200676. [CrossRef]
51. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
52. Zhao, H.; Qi, X.; Shen, X.; Shi, J.; Jia, J. Icnet for real-time semantic segmentation on high-resolution images. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 405–420.