

Article

Road Extraction of High-Resolution Remote Sensing Images Derived from DenseUNet

Jiang Xin ¹, Xinchang Zhang ^{2,*}, Zhiqiang Zhang ³ and Wu Fang ⁴

¹ Department of Geography and Planning, Sun Yat-Sen University, Guangzhou 510275, China; jiangx35@mail2.sysu.edu.cn

² School of Geographical Sciences, Guangzhou University, Guangzhou 510006, China

³ College of Surveying and Geo-informatics, North China University of Water Resources and Electric Power, Zhengzhou 450046, China; zhangzhiqiang@ncwu.edu.cn

⁴ College of Surveying and Mapping, Information Engineering University, Zhengzhou 450001, China; xiaoyu_lee@whu.edu.cn

* Correspondence: eeszxc@mail.sysu.edu.cn; Tel.: +86-208-411-5103

Received: 25 September 2019; Accepted: 20 October 2019; Published: 25 October 2019



Abstract: Road network extraction is one of the significant assignments for disaster emergency response, intelligent transportation systems, and real-time updating road network. Road extraction base on high-resolution remote sensing images has become a hot topic. Presently, most of the researches are based on traditional machine learning algorithms, which are complex and computational because of impervious surfaces such as roads and buildings that are discernible in the images. Given the above problems, we propose a new method to extract the road network from remote sensing images using a DenseUNet model with few parameters and robust characteristics. DenseUNet consists of dense connection units and skips connections, which strengthens the fusion of different scales by connections at various network layers. The performance of the advanced method is validated on two datasets of high-resolution images by comparison with three classical semantic segmentation methods. The experimental results show that the method can be used for road extraction in complex scenes.

Keywords: high-resolution remote sensing imagery; multi-scale; road extraction; machine learning; DenseUNet

1. Introduction

The traffic road network is one of the essential geographic element of the urban system, which has critical applications in many fields, such as intelligent transportation, automobile navigation, and emergency support [1]. With the development of remote sensing technology and the advancement of remote sensing data processing methods, high temporal and spatial resolution, remote sensing data can provide high-precision ground information and permit the large-scale monitoring of roads. Remote sensing image data has quickly become the primary data source for the automatic extraction of road networks [2]. Automating road extraction plays a vital role in dynamic spatial development. Extracting the road in the urban area is a significant concern for the research on transportation, surveying, and mapping [3]. However, remote sensing images usually have sophisticated heterogeneous regional features with considerable intra-class distinctions and small inter-class distinctions. It is very challenging, especially in the urban area, as many buildings and trees exist, leading to shadow problems and a large number of segmented objects. The shadows of roadside trees or buildings can be observed from high-resolution images. Consequently, it is challenging to obtain high-precision road network information in the automatic extraction of road networks from remote sensing images.

There are many image segmentation methods for these problems by such conventional methods or machine learning algorithms. These methods are mainly divided into two categories: road centerline

extraction and road area extraction. This paper focuses on extracting road areas from high-resolution remote sensing images. The road centerline is a linear element, and the spatial geometry is a line formed by a series of ordered nodes, which is an essential characteristic line of the road. The road centerline is generally obtained from the segmented image of road binary map through morphology or Medial Axis Transform (MAT) [4]. The road area is a kind of surface element. The road area is generated by image segmentation. The different spatial shape structure of boundary lines forms a variety of shape structures of surface elements [5]. Road centerline extraction [6,7] is used to detect the skeleton of the road, while road area extraction [8–13] generates the pixel-level label of the road, and there are some methods to extract the road area [14] while obtaining the road centerline. Huang et al. [8] try to extract road networks from the Ranging (LiDAR) data and light detection. Mnih et al. [9] used the Deep Belief Network (DBN) model to identify road targets in airborne images. Unsalan et al. [10] integrated three modules of road shape extraction module, road center probability detection module, and graphics-based module to extract road network from high-resolution satellite images. Cheng et al. [11] automatically extracted the road network information from complex remote sensing images based on the probability propagation method of graph cut. Saito et al. [12], based on the output of the channel function is put forward a new method of CNN's tabbed semantic segmentation. Alshehhi et al. [13] proposed an unsupervised road segmentation method based on the hierarchical graph. Road area extraction can be divided into pixel-level classification or image segmentation problems. Song et al. [15] proposed a method of road area detection based on the shape indexing feature of the support vector machine (SVM). Wang et al. [16] present a road detection method based on salient features and gradient vector flow (GVF) Snake. Rianto et al. [17] proposed a method to detect main roads from SPOT satellite images. The traditional road extraction method depends on the selected features. Zhang et al. [18] selected the seed points on the road, determined the direction, width, and starting point of the road in this section with a radial wheel algorithm, and proposed a semi-automatic method for road network tracking in remote sensing images. Movaghati et al. [19] proposed a new road network extraction framework by combining an extended Kalman filter (EKF) and a special particle filter (PF) to recover road tracks on obstructed obstacles. Gamba et al. [20] used adaptive filtering steps to extract the main road direction, and then proposed a road extraction method based on the prior information of road direction distribution. Li et al. [21] gradually extracted the road from the binary segmentation tree by determining the region of interest of the high-resolution remote sensing image and representing it as a binary segmentation tree.

However, the manually selected set of features is affected by many threshold parameters, such as lighting and atmospheric conditions. This empirical design method only deals with specific data, which limits its application in large-scale datasets. Deep learning is a representation learning method with multiple levels of representation, which is obtained by combining nonlinear but straightforward modules, each module representing a level of representation to a higher, slightly more abstract level. It allows raw data to be supplied to the machine and representations to be automatically discovered. In recent years, the deep convolutional network has been widely used in solving quite complex classification tasks, such as classification [22,23], semantic segmentation [24,25], and natural language processing [26,27].

Most importantly, these methods have proven to be profoundly robust to the appearance of different images, which prompted us to apply them to fully automated road segmentation in high-resolution remote sensing images. Long promoted the fully-convolutional network (FCN) and applied it to the field of semantic segmentation. Likewise, new segmentation methods based on deep neural networks and FCN were developed to extract roads from high-resolution remote sensing images. Mnih [28] put forward a method that combined the context information to detect road areas in aerial images.

He et al. [29] improves the performance of road extraction networks by integrating the spatial pyramid pool (ASPP) with the Encoder–Decoder network to enhance the ability to extract detailed features of the road. Zhang et al. [30] enhanced the propagation efficiency of information flow by fusing dense connections with convolutional layers of various scales. Aiming at the rich details of remote

sensing images, Li et al. [31] proposed a Y-type convolutional neural network for road segmentation of high-resolution visible remote sensing images. The proposed network not only avoids background interference but also makes full use of complex details and semantic features to segment multi-scale roads. RSRCNN [32] extracts roads based on geometric features and spatial correlation of roads. Su et al. [33] enhanced the U-Net network model based on available problems. According to the characteristics of a small sample of aerial images, Zhang et al. [34] proposed an improved network-based road extraction design framework. By refining the CNN architecture, Gao et al. [35] proposed the refined deep residual convolutional neural network (RDRCNN) to enable it to detect the road area more accurately. To solve the problems of noise, occlusion, and complex background, Yang et al. [36] successfully designed an RCNN unit and integrated it into the U-Net architecture. The significant advantage of this unit is that it retains detailed low-level spatial characteristics. Zhang et al. [37] proposed the ResU-Net to extract road information by combining the advantages of a residual unit and U-Net. According to the characteristics of the narrow, connected, complex road, Zhou et al. [38] proposed the D-LinkNet model while maintaining the road information, integration of the multi-scale characteristics of the high-resolution satellite images. Based on the iterative search process guided by the decision function of CNN, Bastani [39] proposed RoadTracer, which can automatically construct accurate road network maps directly from aerial images. For irregular footprint problems between road area and image, Li et al. [40] proposed a combining GANs and multi-scale context polymerization of semantic segmentation method, used for road extraction of UAV remote sensing images. Xu et al. [41] put forward a kind of road extraction method based on local and global information, to effectively extract the road information in remote sensing images.

Inspired by the Densely Connected Convolutional Networks and U-Net, we propose the DenseUNet, an architecture that takes advantage of Densely Connected Convolutional Networks and U-Net architecture. The proposed deep convolutional neural network is based on the U-Net architecture. There are three differences between our deep DenseUNet and U-Net.

First, the model used dense units rather than ordinary neural units as the basic building blocks. Second, the proportion of road and non-road in remote sensing images is seriously unbalanced. Thus, this paper tries to analyze and propose ideas in terms of this issue. Finally, the performance of the proposed method is validated by comparison with three classical semantic segmentation methods.

2. Methods

2.1. Encoder–Decoder Architecture

State-of-the-art semantic image segmentation methods are mostly based on Encoder–Decoder architecture such as FCN [42], U-Net [43], SegNet [44]. An end-to-end trainable neural network recognizes the road in images and accurately segmented at the pixel level. Encoders usually use pre-trained models (such as VGG, Inception and ResNet), and each encoding layer includes the convolution, batch normalization (BN), the ReLU function and max pool layer. Each convolutional layer extracts features from all the maps in the previous layer, which has characteristics of simple structure and strong adaptability. Batch normalization [45] normalizes the input of each layer to reduce the internal-covariate-shift problem. It accelerates training and acts as a regularizer. The result shows that estimators based on a connected deep neural network with ReLU activation function and correctly selected the network. Pooling layer aims to compress the input feature map, which reduces the number of parameters in the training process and the degree of overfitting of the model. The main task of the Decoder is to map the distinguishable features to the pixel space for dense classification. Road network density refers to the ratio of the total mileage of the road network to the space of a given area. For the extraction of relatively dense urban roads (in the same area, there are more roads), especially from high-resolution images, significant obstacles are leading to unreliable extraction results: complex image scenes and road models, as well as occlusion caused by high buildings and their shadows. Because of the above problems, this paper proposes DenseUNet, which is also based on

Encoder–Decoder architectures and designs a more dense connection mechanism for the Encoder layer. Because of the complexity of road scenes, U-Net cannot identify road features at a deeper level, and the generalization ability of multi-scale information is limited, which cannot adequately convey scale information. DenseUNet is a network architecture in which each layer feeds forward (within each dense block) directly to each of the other layers. For each layer, the feature map for all other layers is treated as a separate input, and its feature map is passed as input to all subsequent layers. Additionally, our approach has far fewer parameters due to the intelligent construction of the model. This kind of network design method not only extracts low-level features such as road edges and textures but also identifies the deep contour and location information of the road.

2.2. Backpropagation to Train Multilayer Architectures

Multilayer architectures can be trained by stochastic gradient descent. If only the input function and internal weight of the module are relatively smooth, the gradient can be computed by using the backpropagation process. The backpropagation process used to compute the gradient of the objective function about the weight of stacked multilayer modules is only the practical application of chain rules of derivatives. The significant idea is that the derivative (or gradient) of the module input can be computed by working backward from the gradient of the module output [46].

Figure 1 shows that the input space becomes iteratively warped until the data points become distinguishable through the data flow at various layers of the system. In this way, it can learn highly complex functions. Deep learning is a form of presentation learning—providing the machine with the raw data and developing the representations needed for its pattern recognition—that consists of multiple representation layers. These layers are usually arranged sequentially and consist of a large number of original nonlinear operations, where the representation of such a layer (the original data input) is fed into the next layer and converted to a more abstract representation [47]. The output layer uses softmax activation function to classify the image in one of the classes, and we can use fine-tuned CNNs as feature extractors to achieve better results.

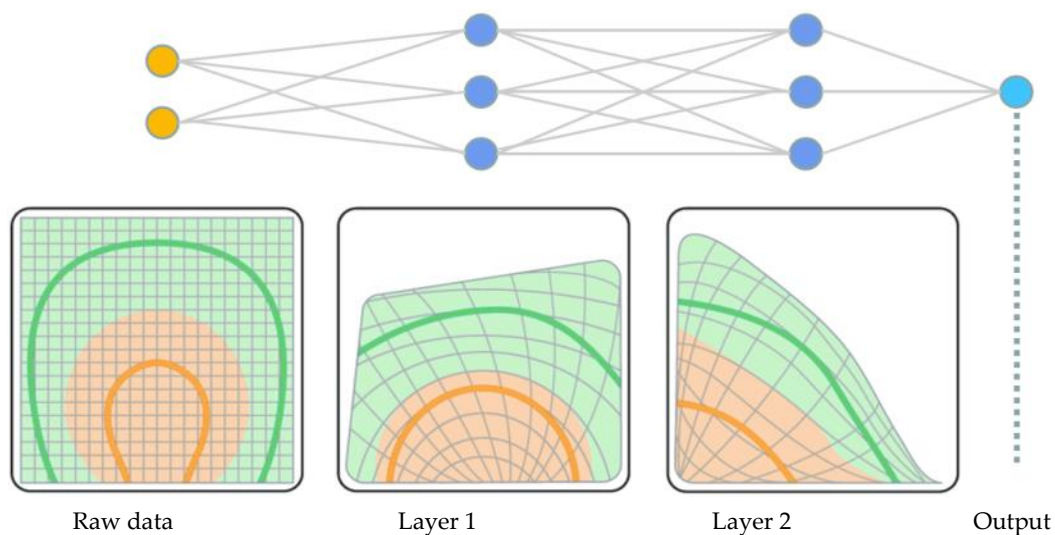


Figure 1. When data flows from one layer to another of the neural network, they are linearly separated by iteratively distorting the data. The final output layer outputs the probabilities of any class. This example illustrates the basic concepts of large-scale network usage.

2.3. DenseUNet

2.3.1. Network Architecture

We chose U-Net as the primary network architecture. In semantic segmentation, in order to achieve better results, it is essential to retain low-level details while acquiring high-level semantic

information. The low-level features can be copied to the corresponding high-level to create information transmission paths, allowing signals to propagate naturally between the lower and higher levels, which not only helps the backpropagation in the training process but also compensates for the low-level and details of the high-level semantic features. We show that making use of dense units instead of ugly units can further improve the performance of U-Net. In this paper, the dense block is used as sub-module for feature extraction. By design, DenseUNet allows the layer to access all of its previous feature maps. DenseUNet takes advantage of the potential of the network to efficient compression models through feature reuse. It encourages reuse of features throughout the network and leads to a more compact model.

To restore the spatial resolution, FCN introduces an up-sampling path that includes convolution, up-sampling operations (transpose convolution or linear interpolation), and skip connections. In DenseUNet, we replace the convolution operation with up-sampling operations and transform it. The transition up module consists of a transposed convolution, which upsamples the previous feature mapping. Then the up-sampling feature map is connected to the input from the encoder skip connection to form a new input. We utilize an 11-level deep neural network architecture to extract road areas, as shown in Figure 2.

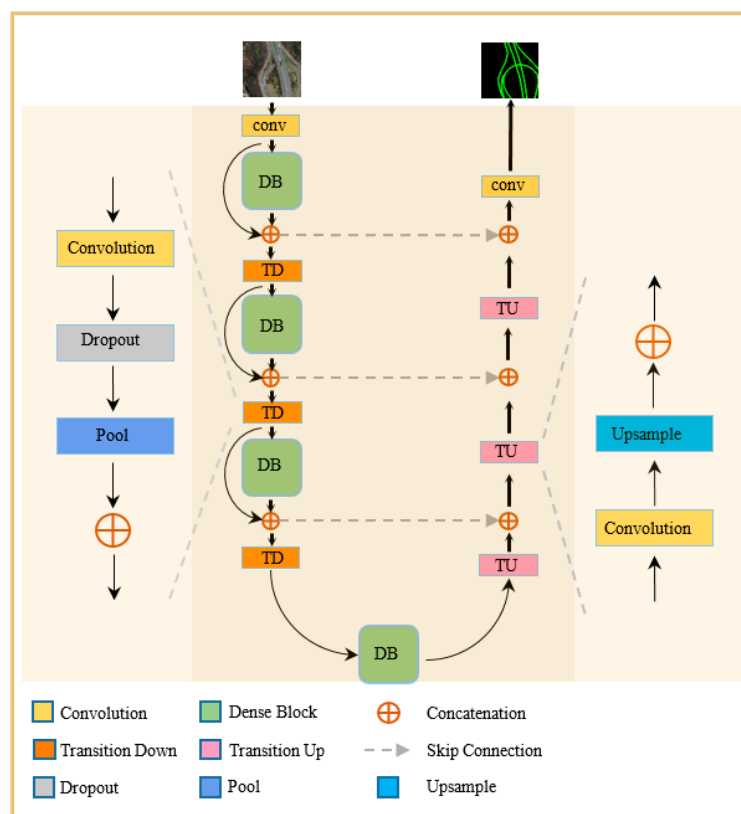


Figure 2. The architecture of the proposed deep DenseUNet. The dense block takes advantage of the potential of the network to efficient compression models through feature reuse.

2.3.2. Dense Block

Deep neural networks extract multi-level features of remote sensing images from low to high by convolution and pooling operations. The first few layers of convolution neural networks mainly extract low-level features such as road edges and textures, while deep-level networks extract features more complete, including road contours and location information. It can improve the performance of multi-layer neural networks and extract higher-level semantic information; however, it may hinder training and cause degradation problems. This is a problem with backpropagation [48]. He et al. [49]

proposed residual neural networks to speed up training and solve degradation problems. The residual neural network consists of a series of residual units. Each unit can be represented in the following form:

$$Z_l = H_l(Z_{l-1}) + Z_{l-1} \tag{1}$$

Among them, Z_{l-1} and Z_l are the input and output of the l^{th} residual unit, and $H_l(\cdot)$ is the residual function. Therefore, for ResNet model, the output of the l^{th} layer is the composition of the $l-1^{th}$ identity mapping and the $l-1^{th}$ nonlinear transformation. The connection between the low-level and the high-level of the network will facilitate the dissemination of information without degradation. However, this kind of integration destroys the information flow between the layers of the network to a certain extent [50]. Here, we present the DenseUNet, a semantic segmentation neural network that combines the advantages of a densely concatenated convolutional network and U-Net. This architecture can be considered an extension of ResNet, which iteratively sums up the previous feature mappings. However, this small change has some exciting implications: (1) feature reuse, all layers can easily access the previous layer, so that the information in previously computed feature map can be easily reused; (2) parameter efficiency, DenseUNet is more effective in parameter usage; (3) implicit in-depth supervision, because of the short-path of all feature graphs in the architecture, DenseUNet provides deep supervision. Figure 3 is the basic dense network unit in this paper.

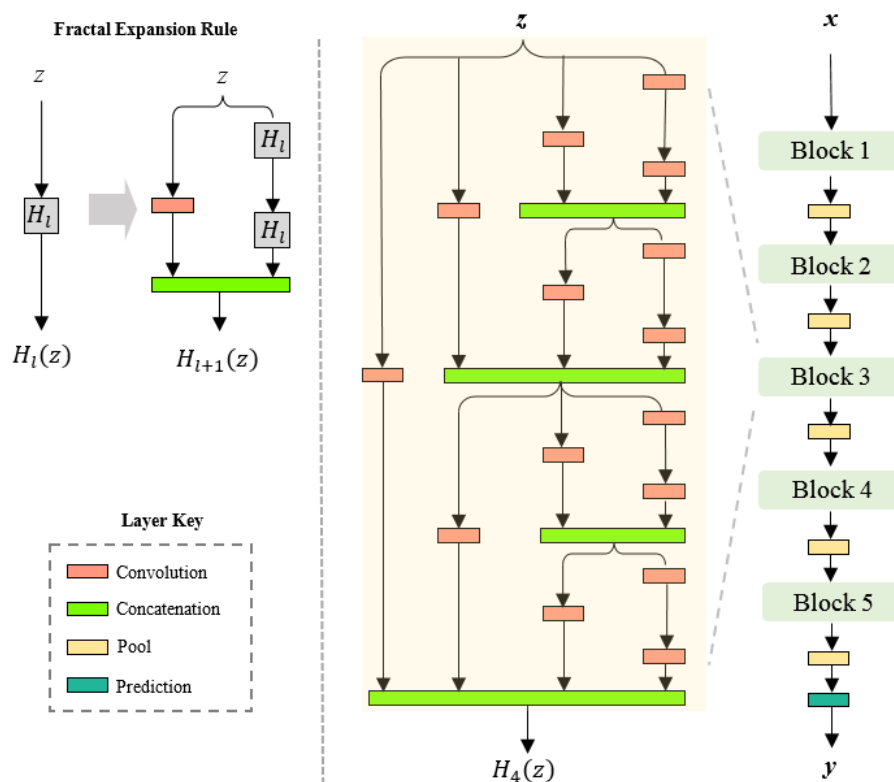


Figure 3. Dense network unit. Fractal structures have statistical or similar self-similar forms.

Dense network elements are fractal architectures. **Dense block layers are connected to each other so that each layer in the network accepts the characteristics of all its previous layers as input.** Left: simple extended rules generate fractal architectures with l intertwined columns. Basically, $H_1(Z)$ has a single layer of the selected type (e.g., convolution) between input and output. The connection layers compute the average value of element-wise. Right: Deep convolution neural network reduces spatial resolution periodically by pooling. A fractal version uses $H_1(Z)$ as the building block between pooling layers. A block such as Stack B produces a network whose total depth (measured as a

convolutional layer) is $B \times 2^{C-1}$. Dense units consist of three parts: dense connection, growth rate, and bottleneck layers.

Dense connections. In order to further enhance the transmission of information among network layers, this paper constructs a different connection mode: by introducing direct connections from any layer to all subsequent layers. Figure 3 shows the layout of DenseUNet. Consequently, the Z_l layer receives the feature-maps of all other layers. Z_0, Z_1, \dots, Z_{l-1} , as input:

$$Z_l = H_l([Z_0, Z_1, \dots, Z_{l-1}]) \quad (2)$$

Among them $[Z_0, Z_1, \dots, Z_{l-1}]$ refers to the series of features generated in layer $0, \dots, l-1$. To promote implementation, the multiple inputs of $H_l(\cdot)$ in Equation (2) are concatenated into a single tensor. We define $H_l(\cdot)$ as a composite function of three continuous operations: batch normalization, followed by a 3×3 convolution and a rectified linear unit.

Growth rate. H_l generates G feature-maps, and then the l th layer has $G_0 + G \cdot (l-1)$ input feature maps, where G_0 is the number of channels in the input layer. The difference between DenseUNet and existing network architectures is that DenseUNet can have skinny layers. The hyper-parameter G is called the growth rate of the network.

Bottleneck layers. Although each layer generates only G output element mappings, it usually has more inputs. Literature [51] has noticed that before each 3×3 convolution, 1×1 convolution can be introduced as the bottleneck layer to reduce the number of input feature maps and improve the computational efficiency. We utilize such a bottleneck layer to refer to our network, i.e., the BN-Conv-ReLU version for H_l . Figure 4 shows the operation of dense block layers, transition down and transition up.

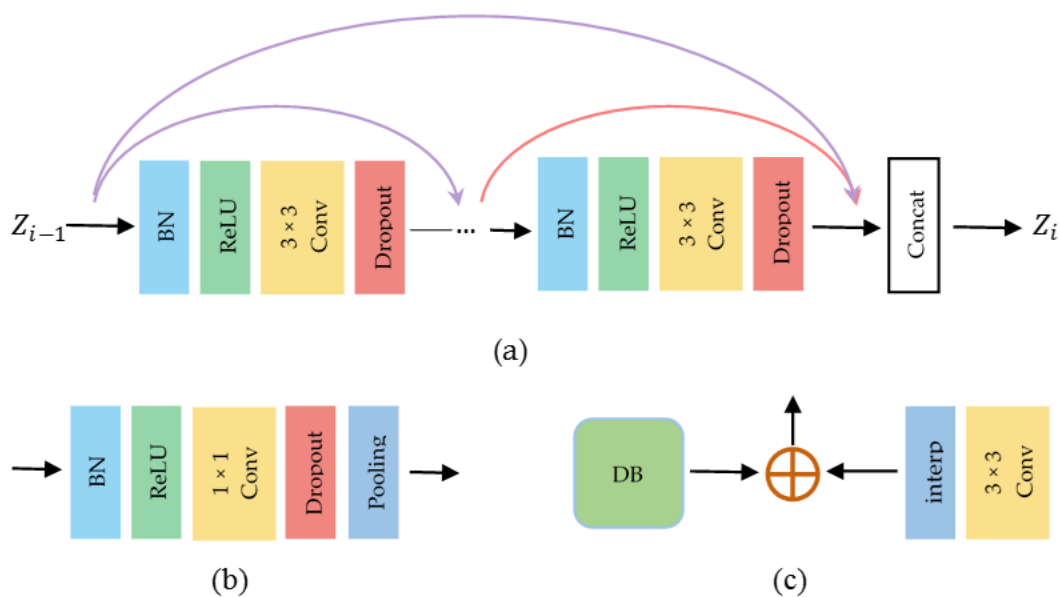


Figure 4. Basic layers of dense block, Transition Down, and Transition Up. (a) The dense block layer consists of BN, followed by ReLU and dropout; (b) Transition Down consists of BN followed by ReLU, dropout and a max-pooling of size 2×2 ; (c) Transition Up consists of a convolution, using nearest-neighbor interpolation to compensate for the loss of pooling process spatial information.

In our experiments on Conghua roads dataset and Massachusetts roads dataset, we used DenseUNet structure with five dense blocks on 256×256 input images. The number of feature maps in other layers also follows the setting G . In the present study, and we used Adam optimizer to minimize the classification cross-entropy. Let Y be a reference foreground segmentation with values y_i , and X be a prediction probability map of the foreground markers on the N image elements x_i , where the

probability of background class is $1 - x_i$. The cross-entropy represents the dissimilarity between the approximate output distribution and the real distribution of the labels. The cross-entropy describes the difference between the true distribution of the input data and the distribution of the model obtained through training. The binary cross-entropy loss function is defined as:

$$loss = -\frac{1}{N} \sum_i^N (y_i \cdot \log x_i + (1 - y_i) \cdot \log(1 - x_i)) \quad (3)$$

The reasonable ratio of positive and negative samples is about 1:1 for feature selection in binary classification tasks. However, we find that the serious class imbalance between foreground and background is the central cause of high-resolution remote sensing images in the training process of semantic segmentation.

When the loss function gives equal weight to positive and negative samples, the category with large sample dominates the training process, and the training model is inclined to the category with a large sample, which reduces generalization ability of the model. We suggest reshaping the standard cross-entropy loss to solve the class imbalance problem in order to reduce the loss assigned to large samples. The weighted cross-entropy form of two-class can be expressed as:

$$loss = -\frac{1}{N} \sum_i^N (\theta_1 \cdot y_i \cdot \log x_i + (1 - y_i) \cdot \log(1 - x_i)) \quad (4)$$

where θ_1 is attributed to the weight of the foreground class, here defined as:

$$\theta_1 = \frac{N - \sum_i^N x_i}{\sum_i^N x_i} \quad (5)$$

By appropriately increasing the loss caused by the fault positive samples, the problem of the vast difference between the positive and negative samples is solved to some extent.

3. Experiments

3.1. Model Preprocessing

3.1.1. Software and Hardware Environment

In order to examine the proposed method, we construct a system platform, which is mainly composed of two parts: the software and hardware environment. The training and testing of deep neural networks require high-performance machines, which consumes a lot of video memory during training. TensorFlow is provided with the advantages of high efficiency, strong expansibility, and high flexibility design, and with the support of TensorFlow researchers, the efficiency of TensorFlow is improved. Based on the above reasons, this paper selects TensorFlow framework for network training. The basic configuration is shown in Table 1:

Table 1. Platform configuration.

Hardware		Software					
memory	hard disk	CPU	GPU	OS	CUDA	TensorFlow	python
16GB	1TB	Core-i7-8700K	GTX1080Ti	Ubuntu16.04	9.0	1.5	2.7.12

3.1.2. Data Augmentation

The deep learning model is trained with sufficient data, with the increase of the input size of the deep neural network, the training parameters after convolution operation also increase. In order to make use of the video memory and increased training efficiency, we utilize a 256×256 window to crop image blocks. One of the main problems in such models and signature verification systems is the low number of samples for training the model. Although transfer learning is effective in other domains, the remote sensing images are essentially different from traditional images by rich spectral setting, a wide range of image values, and different color and texture distributions. The image enhancement method is introduced to improve the generalization ability of the model. The deep learning method uses the method to add more data to the training dataset, which is called data augmentation. Data augmentation has already proved to bring many benefits to convolutional neural networks (CNNs) [52]. For example, as a regularizer, it is used to prevent overfitting in neural networks [53] and to improve the performance of unbalanced class problems [54]. As shown in Figure 5, the training set is expanded by six times.

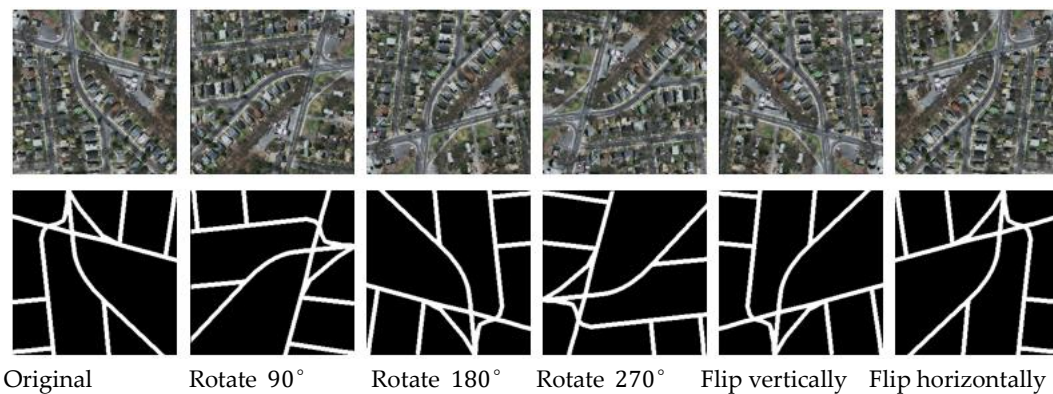


Figure 5. Data augmentation. The method mainly includes rotation, flipping (horizontally and vertically), and cropping operations.

3.1.3. Hyper-Parameters Selection

The process of searching optimal models requires parallel training of multiple models. The selection of learning batch size, learning rate, and optimization algorithm makes the model unique and different from other models. The process of selecting the best model requires the hyper-parameters to be optimized. We use TensorFlow to perform parallel data training with many models. Three hyper-parameters batch size (batch size, learning rate, and epochs) allow parallel training of multiple models, and the accuracy of test datasets determines the best model. We have studied various methods to enable deep learning models to be learned from the training dataset. We studied various methods to learn deep learning models from training data sets. Hyper-parameters can be used to activate the training process. Adam is an adaptive learning method that requires less tuning, is computationally efficient, and is superior to other stochastic optimization methods. The network hyper-parameter settings are shown in Table 2. We chose Adam as the optimization method, and it represents faster convergence than the standard stochastic gradient with momentum. We fix the parameters of Adam as recommended in Reference [55]: $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

Table 2. Hyper-parameters.

Hyper-Parameter	Grid Search
batch size	(2, 4, 8, 16)
epochs	(50, 100, 150, 200)
learning rates	(1×10^{-9} , 1×10^{-5} , 1×10^{-3} , 1×10^{-1})

Compared with the classical U-Net, SegNet, GL-Dense-U-Net, and FRRN-B network, we evaluated the proposed method on two urban scenario datasets: Conghua road dataset, and Massachusetts road dataset. For the sake of quantitatively estimate the performance of the semantic segmentation method, we show the precision, recall, F1-Score, intersection over union (IoU) and kappa as different metrics for performance. The recall rate is defined as the ratio of the correct detection category to the correct detection category and the sum of a false negative, which will be used to assessments of the road integrity. The precision rate is the proportion of successes made by a classifier over the whole instance set, which reflects on the correctness of the road. The F1-score is the harmonic average of precision and recall, computed based on the number of errors detected by computers and manual evaluators. The Intersection over Union (IoU) is only the ratio of the overlap area between the truth and predicted regions of interest on the ground to the area surrounded by them. The kappa coefficient is a statistic which measures inter-rater agreement for specific items, and it is generally used to assess the accuracy of remote sensing image classifications.

3.2. Massachusetts Dataset

The Massachusetts dataset [56] has an image resolution of 1 m, and each image contains $3 \times 1500 \times 1500$ pixels. The open road dataset contains 1711 aerial images with a total area of more than 2600 square kilometers. The dataset is divided into 1108 training images, 14 validation images, and 49 test images. Figure 6 shows that U-Net, SegNet, and FRRN-B models can correctly identify most of the roads. Although these models eliminate the effects of shadows and buildings to a certain extent, the extraction results show that the correctness of intensive road is lower than in other regions. The results of these models are poorly continuous, and the edge of the road is not distinct enough. U-Net and SegNet performed poorly and lack of the necessary connectivity in the intensive road. From the sixth and seventh columns, the performance ability of GL-Dense-U-Net is equal to that of DenseUNet. Both models show good results in both single lane and dual lanes

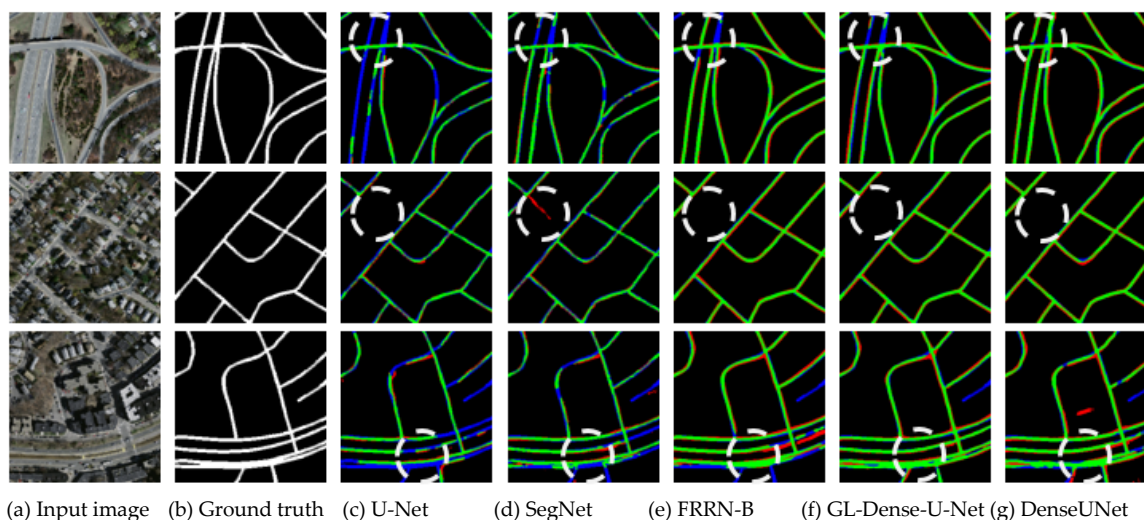


Figure 6. Images of the original actual color composite image are displayed and classified in three regions using deep learning methods. True positive (TP), false negative (FN) and false positive (FP) were marked as green, blue, and red, respectively.

3.3. Conghua Dataset

The image resolution of Conghua dataset is 0.2 m, which consists of three bands: Red, Green, and Blue (RGB). There are 47 aerial images in this dataset, and each image consists of $3 \times 6000 \times 6000$ pixels. Among these, 80% of the data is used for training, and the remaining 20% data is used for model validation. Figure 7 shows that the white dotted line area is covered with thick trees, especially in urban environments, where model performance is more challenging than other areas, and the road

occlusion is more frequent due to trees. The method we propose is hardly affected by shadow occlusion, and the average performance is better than the other three classical semantic segmentation algorithms based on convolution neural network. The performance of the GL-Dense-U-Net model on this data set is comparable to that of DenseUNet, and the extracted road edge information is relatively complete, which maintains functional connectivity. We can extract the local feature information of the image accurately and effectively. Figure 8 shows the details of the shaded area.

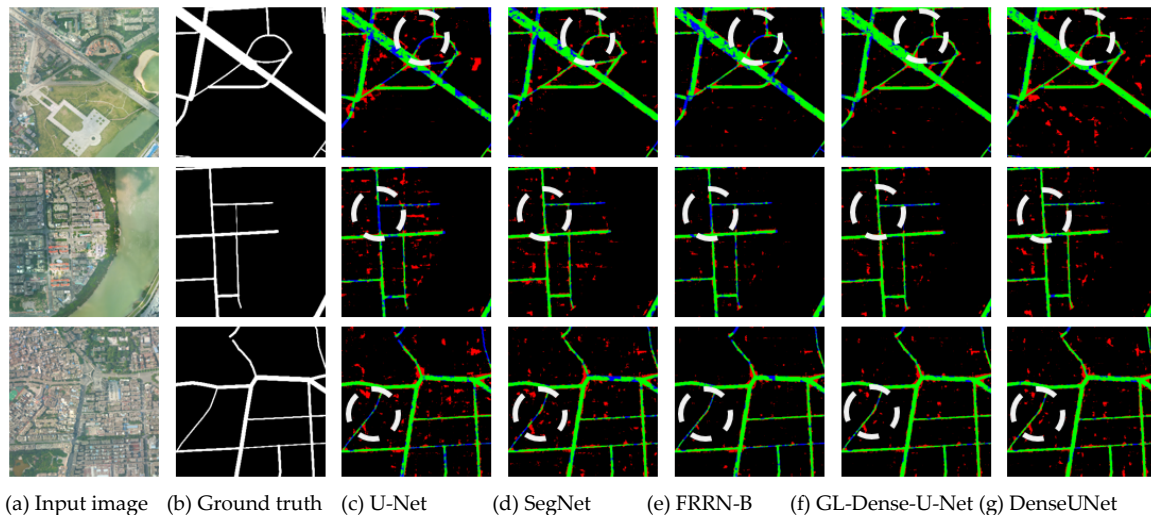


Figure 7. Images of the original actual color composite image are displayed and classified in three regions using deep learning methods. True positive (TP), false negative (FN) and false positive (FP) were marked as green, blue, and red, respectively. The white dotted line in the images is enlarged for close-up inspection in Figure 7.

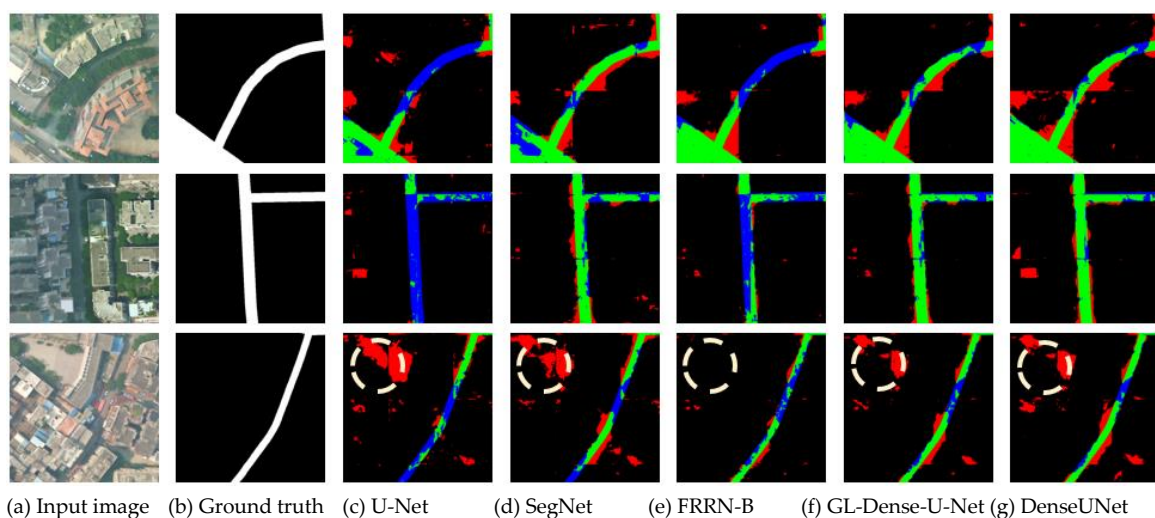


Figure 8. A close-up view of the original true-color composite image and classification results is displayed across three regions using the deep learning method. The images are the subset from the white dotted line marked in Figure 7. True positive (TP), false negative (FN) and false positive (FP) were marked as green, blue, and red, respectively.

3.4. Accuracy Evaluation

Table 3 shows a comparison of the accuracy of automatic classification. We find that the proposed method achieves the highest accuracy, and both F1-score and kappa are significantly higher than three classical semantic segmentation methods on both datasets. The kappa metrics for the classification results were 0.703 and 0.801, respectively. The proposed method provides the most important value

for the F1-score, which involves recall and accurate metrics. The experimental results show that the average performance of the method in recall rate, accuracy, and F1-score is better than the other three classical semantics segmentation methods. In addition, it was found that the method can produce the relatively high average performance of IoU, and kappa over all the images in the test set, which is consistent with the predicted results of Figures 6 and 8.

Table 3. The experimental results of road extraction.

Model	M-Dataset					C-Dataset				
	P (%)	R (%)	F1 (%)	IoU (%)	Kappa	P (%)	R (%)	F1 (%)	IoU (%)	Kappa
U-Net	58.92%	70.81%	60.78%	70.91%	63.65%	84.29%	75.23%	71.83%	77.35%	77.43%
SegNet	61.35%	71.33%	62.64%	71.91%	65.41%	85.03%	77.04%	73.94%	78.83%	78.52%
FRRN-B	76.51%	64.87%	66.71%	74.22%	67.70%	83.92%	77.22%	73.62%	78.72%	78.16%
GL- Dense-U-Net	78.48%	70.09%	73.98%	72.73%	70.19%	85.33%	79.07%	76.41%	80.67%	80.35%
DenseUNet	78.25%	70.41%	74.07%	74.47%	70.32%	85.55%	78.51%	76.25%	80.89%	80.11%

Figures 6 and 8 illustrate three example results of U-Net, SegNet, FRRN-B, GL-Dense-U-Net, and the proposed DenseUNet. The results show that compared with the other four methods, our method has the advantages of high accuracy and low noise. Especially in the case of dense roads and shadows, our method can divide each lane with high reliability and get prominent shadows, as shown in the third row of Figures 6 and 8.

3.5. Model Analysis

Road background information is essential when analyzing complex structured objects. Our network takes into account the information around the road to facilitate the distinction between roads and similar objects, such as building roofs and dense trees. The context information is robust when the road is occluded. From the first row of Figure 7, some of the roads in the circle are covered by trees. Three classical semantics segmentation methods cannot detect the road under the tree; however, our method has successfully marked them to some extent. A case of failure is shown in the gold dotted line of Figure 8; the proposed method has a distinct error detection rate in impervious surface. It is mainly because most of the roads in the urban impervious surface are not labeled. Therefore, considering that our network regards them as contextual information of the foreground, these roads share the same characteristics as normal roads. We provide a better insight into the performance of the proposed method. In Figure 9, we show the loss and performance curves during system training. The loss of the four models slowly decreases as the training time increases and eventually stabilizes. Although the U-Net model showed large changes in the initial stage of the model training, it finally reached a convergence state. It can be seen that the improved model ultimately achieves good convergence. The connections in dense units and skipping connections between the lower and higher levels of the network help to spread information without degradation, so that a neural network with fewer parameters can be designed; however, better comparability can be achieved in semantic segmentation performance.

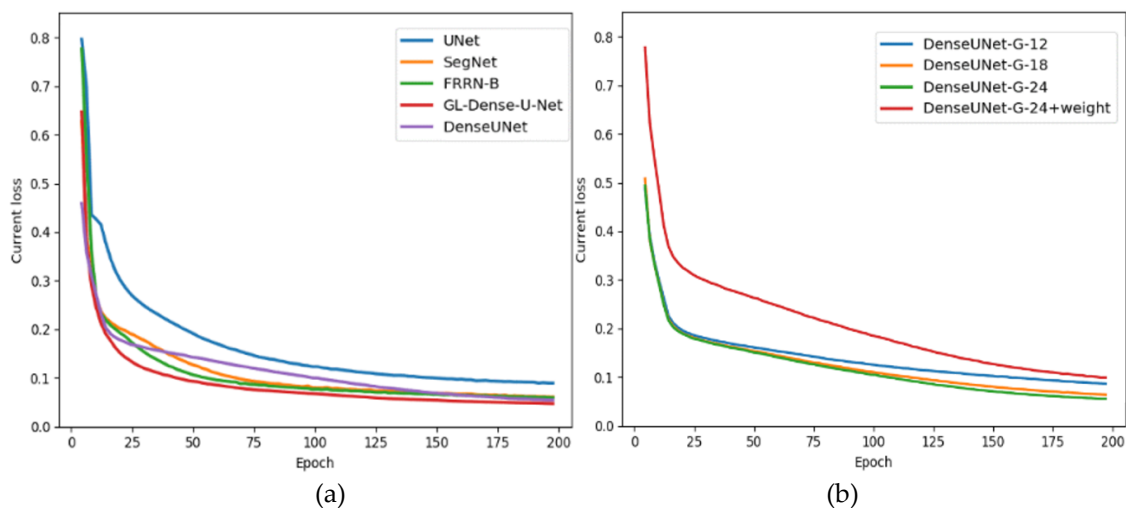


Figure 9. Loss of training. (a) The five curves of blue, yellow, green, red and purple represent the losses of U-Net, SegNet, FRRN-B, GL-Dense-U-Net, and DenseUNet; (b) The four curves represent models with different growth rates and modified weights

DenseUNet extracts multi-level features from different stages of the dense block, which strengthens the fusion of different scales. We train DenseUNet with different growth rates, G . The main results on two sets of data dataset are shown in Table 4. It can be seen from the accuracy that the model has the best performance (when the parameter G is equal to 24). Besides, Table 4 shows that relatively small growth rates are sufficient to achieve excellent results on the test datasets. The growth rate defines the amount of new information provided by each layer for the global state. It can be accessed from anywhere in the network and does not need to be replicated between layers in traditional network architecture.

Table 4. Results of different growth factors.

G	M-Dataset			C-Dataset		
	OA (%)	IoU (%)	Kappa (%)	OA (%)	IoU (%)	Kappa (%)
12	92.22%	73.24%	69.55%	94.18%	73.24%	69.55%
18	93.13%	74.04%	70.11%	94.87%	74.04%	70.11%
24	93.93%	74.47%	70.32%	95.02%	80.89%	70.32%

4. Discussion

Table 5 shows the statistics of the deep learning model and the variations of DenseUNet. The average running time was calculated by iterating 50 times. U-Net adopts a shallow Encoder–Decoder structure, which requires less computational resources and less reasoning time than other models. However, the road integrity extracted from two sets of data is sparse. DenseUNet adopts a custom encoder–decoder architecture, so it maintains a balance between computing resources and reasoning time. It consumes less computing resources and reasoning time than other models.

Table 5. Compare the network efficiency between the tested deep learning model and DenseUNet.

Model	Inference (ms)	Model Size (MB)	FPS
U-Net	340	106	32
SegNet	204	419	33
FRRN-B	338	297	20
GL-Dense-U-Net	1152	1690	22
DenseUNet-G-12	316	118	23
DenseUNet-G-18	450	279	17
DenseUNet-G-24	472	514	15

GL-Dense-U-Net is equivalent to DenseUNet in terms of various indicators. GL-Dense-U-Net consists of Local Attention Units (LAU) and Global Attention Units (GAU). The 1×1 , 3×3 , 5×5 , 7×7 kernels are respectively used for convolution operation by LAU, and finally integrated step by step from the bottom to the top. GAU introduces global average pool (GAP) into the unit to extract comprehensive road information. However, since the GL-Dense-U-Net encoding and decoding layers are composed of dense unit blocks provided by DenseNet, and LAU unit (the feature graph of different scale is fused to realize the attention of pixel-level information) is added in the encoding stage while GAU unit (feature maps from low and high levels are considered, and global information is provided to restore features) is connected later in the decoding stage, the GL-Dense-U-Net model is the largest and the inference time is the longest. DenseUNet adopts dense unit modules in the coding stage, while the sampling stage in the decoding layer adopts the skip connection characteristic of U-Net. Therefore, DenseUNet requires less inference time of 316 ms and a smaller model size of 118 MB than other models. In general, DenseNet is more effective than most models. On the other hand, G feature maps are output after the convolution of all layers in the dense block. The model sets a small growth rate ($G = 12$) to get good results, as shown in Table 4. The overall accuracy and the mIoU of DenseUNet-G-12 in Massachusetts datasets reached 92.22% and 73.24% respectively

In order to further verify the reliability of the proposed model, two groups of remote sensing image data with different resolutions were selected to compare four classical image segmentation models. In Massachusetts datasets, the overall accuracy and the mIoU of DenseUNet in the Massachusetts dataset achieved 93.93% and 74.47%, respectively. The Conghua dataset achieved 95.02% and 80.89%, respectively. In particular, the classification result of Massachusetts is better than that of GL-Dense-U-Net [41]. In general, DenseUNet performs better than Massachusetts datasets in Conghua datasets, which may be a higher data resolution from the dataset.

The developed DenseUNet has excellent potential for improvement. First, the smoothness of road contour is a key factor that affects the accuracy of road extraction. In the two sets of prediction datasets, we found that compared with the ground truth value, the predicted result road had information loss of edge and contour. Obtaining accurate road profile information is still a challenging task. Second, different network models are suitable for different scenarios, such as PSPNet [57], DeepLabv3+ [58], and BiSeNet [59], etc., which are suitable for real-time segmentation of street view. It is usually necessary to design the network according to specific tasks to obtain the best performance. Neural Architecture Search (NAS) is a kind of automated neural network design technology, which can automatically design high-performance network structure according to the sample set through the algorithm. This architecture can effectively reduce the use and implementation cost of the neural network. Third, we only focused on the performance of different deep learning models during the experiment. Traditional methods, such as threshold-based methods and object-based methods [60], have not been compared, and a more comprehensive comparison of these methods is needed in the future.

5. Conclusions

We propose an efficient road extraction method based on a convolution neural network for high-resolution remote sensing images. The model combines the virtue of dense connection mode and U-Net and solves the problem of tree and shadow occlusion to a certain extent, which we call DenseUNet. In particular, we use a U-Net architecture combined with a suitable weighted loss function to place more emphasis on foreground pixels. Following simple connection rules (fractal extensions), DenseUNet naturally integrates deep supervision, the properties of identity mappings, and diversified depth attributes. The dense connections within dense units and the skip connections between the encoding and decoding paths of the network will help to transfer information and accelerate computation, so they can learn more compactly and get more accurate models.

Although deep neural networks have acquired remarkable success in many fields, there are no sophisticated theories yet. However, one of the critical disadvantages of deep learning models is their limited interpretability, and often these models are described as “black boxes” that do not provide insight into their inner workings. On the other hand, it will be challenging to create a general model through theoretical guidance. Hence, the results obtained from such specific planning problem are difficult to apply to other problems in the same field. We plan to use the trained DenseUNet model to transfer knowledge to improve new tasks in future work.

Author Contributions: J.X. and X.Z. conceived and designed the method; J.X. performed the experiments and wrote the manuscript; Z.Z. provided the Conghua dataset; W.F. revised the manuscript.

Funding: This research was funded by National Key R&D Program of China (Grant No. 2018YFB2100702), National Natural Science Foundation of China (Grant Nos. 41875122, 41431178, 41801351 and 41671453), Natural Science Foundation of Guangdong Province: 2016A030311016, Research Institute of Henan Spatio-Temporal Big Data Industrial Technology: 2017DJA001, National Administration of Surveying, Mapping and Geoinformation of China (Grant No. GZIT2016-A5-147), Hunan Botong Information Co.,Ltd.: BTZH2018001.

Acknowledgments: The authors would like to thank the anonymous reviewers for their constructive comments. The authors are also grateful to Hinton et al. for providing the Massachusetts dataset.

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

References

1. Zhang, Z.; Zhang, X.; Sun, Y.; Zhang, P. Road Centerline Extraction from Very-High-Resolution Aerial Image and LiDAR Data Based on Road Connectivity. *Remote Sens.* **2018**, *10*, 1284. [[CrossRef](#)]
2. Zhou, T.; Sun, C.; Fu, H. Road Information Extraction from High-Resolution Remote Sensing Images Based on Road Reconstruction. *Remote Sens.* **2019**, *11*, 79. [[CrossRef](#)]
3. Bong, D.B.; Lai, K.C.; Joseph, A. Automatic Road Network Recognition and Extraction for Urban Planning. *Int. J. Appl. Sci. Eng. Technol.* **2009**, *5*, 209–215.
4. Miao, Z.; Shi, W.; Zhang, H.; Wang, X. Road centerline extraction from high-resolution imagery based on shape features and multivariate adaptive regression splines. *IEEE Geosci. Remote Sens. Lett.* **2012**, *10*, 583–587. [[CrossRef](#)]
5. Li, Z.; Huang, P. Quantitative measures for spatial information of maps. *Int. J. Geogr. Inf. Sci.* **2002**, *16*, 699–709. [[CrossRef](#)]
6. Liu, B.; Wu, H.; Wang, Y.; Liu, W. Main road extraction from zy-3 grayscale imagery based on directional mathematical morphology and vgi prior knowledge in urban areas. *PLoS ONE* **2015**, *10*, e0138071. [[CrossRef](#)]
7. Sujatha, C.; Selvathi, D. Connected component-based technique for automatic extraction of road centerline in high resolution satellite images. *EURASIP J. Image Video Process.* **2015**, *2015*, 8. [[CrossRef](#)]
8. Huang, X.; Zhang, L. Road centreline extraction from high—Resolution imagery based on multiscale structural features and support vector machines. *Int. J. Remote Sens.* **2009**, *30*, 1977–1987. [[CrossRef](#)]
9. Mnih, V.; Hinton, G.E. Learning to detect roads in high-resolution aerial images. In Proceedings of the European Conference on Computer Vision (ECCV), Crete, Greece, 5–11 September 2010; Daniilidis, K., Maragos, P., Paragios, N., Eds.; Springer: New York, NY, USA, 2010; pp. 210–223.
10. Unsalan, C.; Sirmacek, B. Road network detection using probabilistic and graph theoretical methods. *IEEE Trans. Geosci. Remote Sens.* **2012**, *50*, 4441–4453. [[CrossRef](#)]
11. Cheng, G.; Wang, Y.; Gong, Y. Urban road extraction via graph cuts based probability propagation. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 5072–5076.
12. Saito, S.; Yamashita, T.; Aoki, Y. Multiple object extraction from aerial imagery with convolutional neural networks. *J. Electron. Imaging* **2016**, *2016*, 1–9. [[CrossRef](#)]
13. Alshehhi, R.; Marpu, P.R. Hierarchical graph-based segmentation for extracting road networks from high-resolution satellite images. *ISPRS-J. Photogramm. Remote Sens.* **2017**, *126*, 245–260. [[CrossRef](#)]
14. Cheng, G.; Wang, Y.; Xu, S. Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3322–3337. [[CrossRef](#)]

15. Song, M.; Civco, D. Road extraction using SVM and image segmentation. *Photogramm. Eng. Remote Sens.* **2004**, *70*, 1365–1371. [[CrossRef](#)]
16. Wang, F.; Wang, W.; Xue, B. Road Extraction from High-spatial-resolution Remote Sensing Image by Combining with GVF Snake with Salient Features. *Acta Geod. Cartogr. Sin.* **2017**, *46*, 1978–1985.
17. Rianto, Y.; Kondo, S.; Kim, T. Detection of roads from satellite images using optimal search. *Int. J. Pattern Recognit. Artif. Intell.* **2000**, *14*, 1009–1023. [[CrossRef](#)]
18. Zhang, J.; Lin, X.; Liu, Z.; Shen, J. Semi-automatic road tracking by template matching and distance transformation in urban areas. *Int. J. Remote Sens.* **2011**, *32*, 8331–8347. [[CrossRef](#)]
19. Movaghati, S.; Moghaddamjoo, A.; Tavakoli, A. Road extraction from satellite images using particle filtering and extended Kalman filtering. *IEEE Trans. Geosci. Remote Sens.* **2010**, *48*, 2807–2817. [[CrossRef](#)]
20. Gamba, P.; Dell'Acqua, F.; Lisini, G. Improving urban road extraction in high-resolution images exploiting directional filtering, perceptual grouping, and simple topological concepts. *IEEE Geosci. Remote Sens. Lett.* **2006**, *3*, 387–391. [[CrossRef](#)]
21. Li, M.; Stein, A.; Bijker, W.; Zhang, Q.M. Region-based urban road extraction from VHR satellite images using binary partition tree. *Int. J. Appl. Earth Obs. Geoinf.* **2016**, *44*, 217–225. [[CrossRef](#)]
22. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Twenty-Sixth Annual Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–8 December 2012; pp. 1097–1105.
23. Farabet, C.; Couprie, C.; Najman, L. Learning hierarchical features for scene labeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *35*, 1915–1929. [[CrossRef](#)]
24. Noh, H.; Hong, S.; Han, B. Learning deconvolution network for semantic segmentation. In Proceedings of the IEEE international conference on computer vision (ICCV), Santiago, Chile, 11–18 December 2015; pp. 1520–1528.
25. Li, P.; Zang, Y.; Wang, C. Road network extraction via deep learning and line integral convolution. In Proceedings of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Beijing, China, 10–15 July 2016; pp. 1599–1602.
26. Kim, Y. Convolutional neural networks for sentence classification. *arXiv* **2014**, arXiv:1408.5882.
27. Tang, D.; Qin, B.; Liu, T. Document modeling with gated recurrent neural network for sentiment classification. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP), Lisbon, Portugal, 17–21 September 2015; Lluís, M., Chris, C.-B., Jian, S., Eds.; Association for Computational Linguistics: Lisbon, Portugal, 2015; pp. 1422–1432.
28. Mnih, V. Machine Learning for Aerial Image Labeling. Ph.D. Thesis, University of Toronto, Toronto, ON, Canada, 2013.
29. He, H.; Yang, D.; Wang, S.; Wang, S.Y.; Li, Y.F. Road Extraction by Using Atrous Spatial Pyramid Pooling Integrated Encoder-Decoder Network and Structural Similarity Loss. *Remote Sens.* **2019**, *11*, 1015. [[CrossRef](#)]
30. Zhang, Z.; Wang, Y. JointNet: A Common Neural Network for Road and Building Extraction. *Remote Sens.* **2019**, *11*, 696. [[CrossRef](#)]
31. Li, Y.; Xu, L.; Rao, J.; Guo, L.L.; Yan, Z.; Jin, S. A Y-Net deep learning method for road segmentation using high-resolution visible remote sensing images. *Remote Sens. Lett.* **2019**, *10*, 381–390. [[CrossRef](#)]
32. Wei, Y.; Wang, Z.; Xu, M. Road structure refined CNN for road extraction in aerial image. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 709–713. [[CrossRef](#)]
33. Su, J.; Yang, L.; Jing, W. U-Net based semantic segmentation method for high resolution remote sensing image. *Comput. Appl.* **2019**, *55*, 207–213.
34. Zhang, X.; Han, X.; Li, C.; Tang, X.; Zhou, H.; Jiao, L. Aerial Image Road Extraction Based on an Improved Generative Adversarial Network. *Remote Sens.* **2019**, *11*, 930. [[CrossRef](#)]
35. Gao, L.; Song, W.; Dai, J.; Chen, Y. Road Extraction from High-Resolution Remote Sensing Imagery Using Refined Deep Residual Convolutional Neural Network. *Remote Sens.* **2019**, *11*, 552. [[CrossRef](#)]
36. Yang, X.; Li, X.; Ye, Y.; Lau, R.Y.K.; Zhang, X.F.; Huang, X.H. Road Detection and Centerline Extraction Via Deep Recurrent Convolutional Neural Network U-Net. *IEEE Trans. Geosci. Remote Sens.* **2019**, *59*, 7209–7220. [[CrossRef](#)]

37. Zhang, Z.; Liu, Q.; Wang, Y. Road extraction by deep residual U-Net. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 749–753. [[CrossRef](#)]
38. Zhou, L.; Zhang, C.; Wu, M. D-LinkNet: LinkNet With Pretrained Encoder and Dilated Convolution for High Resolution Satellite Imagery Road Extraction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
39. Bastani, F.; He, S.; Abbar, S.; Alizadeh, M.; Balakrishnan, H.; Chawla, S.; Madden, S.; DeWitt, D. Roadtracer: Automatic extraction of road networks from aerial images. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018.
40. Li, Y.; Peng, B.; He, L.; Fan, K.; Tong, L. Road Segmentation of Unmanned Aerial Vehicle Remote Sensing Images Using Adversarial Network with Multiscale Context Aggregation. *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.* **2019**, *12*, 2279–2287. [[CrossRef](#)]
41. Xu, Y.; Xie, Z.; Feng, Y.; Chen, Z.L. Road extraction from high-resolution remote sensing imagery using deep learning. *Remote Sens.* **2018**, *10*, 1461. [[CrossRef](#)]
42. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), Boston, MA, USA, 7–12 June 2015.
43. 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 (MICCAI), Munich, Germany, 5–9 October 2015.
44. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [[CrossRef](#)] [[PubMed](#)]
45. Normalization, B. Accelerating deep network training by reducing internal covariate shift. *arXiv* **2015**, arXiv:1502.03167.
46. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [[CrossRef](#)] [[PubMed](#)]
47. Esteva, A.; Robicquet, A.; Ramsundar, B. A guide to deep learning in healthcare. *Nat. Med.* **2019**, *25*, 24. [[CrossRef](#)] [[PubMed](#)]
48. Yu, X.; Efe, M.O.; Kaynak, O. A general backpropagation algorithm for feedforward neural networks learning. *IEEE Trans. Neural Netw.* **2002**, *13*, 251–254.
49. He, K.; Zhang, X.; Ren, S. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016.
50. Huang, G.; Liu, Z.; Van, D.M. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
51. Sermanet, P.; Chintala, S.; LeCun, Y. Convolutional neural networks applied to house numbers digit classification. *arXiv* **2012**, arXiv:1204.3968.
52. LeCun, Y.; Boser, B.E.; Denker, J.S. Handwritten Digit Recognition with a Back-Propagation Network. In *Advances in Neural Information Processing Systems 2*; Morgan Kaufmann Publishers: San Francisco, CA, USA, 1990; pp. 396–404.
53. Simard, P.Y.; Steinkraus, D.; Platt, J.C. Best practices for convolutional neural networks applied to visual document analysis. In Proceedings of the International Conference on Document Analysis and Recognition (ICDAR), Edinburgh, Scotland, 3–6 August 2003.
54. Chawla, N.V.; Bowyer, K.W.; Hall, L.O. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357. [[CrossRef](#)]
55. Weinzaepfel, P.; Revaud, J.; Harchaoui, Z. DeepFlow: Large displacement optical flow with deep matching. In Proceedings of the IEEE International Conference on Computer Vision (CVPR), Portland, OR, USA, 23–28 June 2013.
56. Available online: <https://www.cs.toronto.edu/~vmnih/data/> (accessed on 24 October 2019).
57. Zhao, H.; Shi, J.; Qi, X.; Wang, X.G.; Jia, J.Y. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
58. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018.

59. Yu, C.; Wang, J.; Peng, C.; Gao, C.X.; Yu, G.; Song, 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.
60. Baatz, M.; Schäpe, A. Multiresolution segmentation: An optimization approach for high quality multi-scale image segmentation. In *Angewandte Geographische Information—Sverarbeitung*; Strobl, J., Blaschke, T., Griesbner, G., Eds.; Wichmann Verlag: Karlsruhe, Germany, 2000; pp. 12–23.



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).