International Journal of
Geo-Information

Article

Automatic Identification of Overpass Structures: A Method of Deep Learning

Hao Li [1], Maosheng Hu [1,2,*] and Youxin Huang [1]

[1] School of Geography and Information Engineering, China University of Geosciences, Wuhan 430074, China; cug_hao@cug.edu.cn (H.L.); yolandahyx@cug.edu.cn (Y.H.)
[2] National Engineering Research Center for Geographic Information System, Wuhan 430074, China
* Correspondence: humsh@cug.edu.cn; Tel.: +86-134-7610-6137

Received: 22 July 2019; Accepted: 16 September 2019; Published: 18 September 2019

Abstract: The identification of overpass structures in road networks has great significance for multi-scale modeling of roads, congestion analysis, and vehicle navigation. The traditional vector-based methods identify overpasses by the methodologies coming from computational geometry and graph theory, and they overly rely on the artificially designed features and have poor adaptability to complex scenes. This paper presents a novel method of identifying overpasses based on a target detection model (Faster-RCNN). This method utilizes raster representation of vector data and convolutional neural networks (CNNs) to learn task adaptive features from raster data, then identifies the location of an overpass by a Region Proposal network (RPN). The contribution of this paper is: (1) An overpass labelling geodatabase (OLGDB) for the OpenStreetMap (OSM) road network data of six typical cities in China is established; (2) Three different CNNs (ZF-net, VGG-16, Inception-ResNet V2) are integrated into Faster-RCNN and evaluated by accuracy performance; (3) The optimal combination of learning rate and batchsize is determined by fine-tuning; and (4) Five geometric metrics (perimeter, area, squareness, circularity, and W/L) are synthetized into image bands to enhance the training data, and their contribution to the overpass identification task is determined. The experimental results have shown that the proposed method has good accuracy performance (around 90%), and could be improved with the expansion of OLGDB and switching to more sophisticated target detection models. The deep learning target detection model has great application potential in large-scale road network pattern recognition, it can task-adaptively learn road structure features and easily extend to other road network patterns.

Keywords: road network pattern; overpass; deep learning; target detection model; Faster-RCNN

1. Introduction

As the framework of a city, road networks have attracted the concern of scholars in all aspects of urban research and are regarded as one of the five major elements in city construction by Lynch (1960) [1]. The pattern of urban road network is complex and diverse, which reflects the characteristics of road distribution and the political, economic, and cultural characteristics of a city in a particular historical period [2–4]. It is crucial to accurately identify road patterns for road navigation, urban planning, and functional zoning. Therefore, accurately identifying the road patterns in complex road networks have become the focus for GIS research [5].

In urban road networks, the overpass is one of the most important road patterns. An overpass as a three-dimensional cross bridge in a road network refers to a modern building that is built on the intersection of two or more intersecting roads with different levels and different directions. Its main function is to make the vehicles in all directions free from traffic lights at the crossroads so as to pass quickly. Scheiders proposed an identification method for an overpass based on road attributes [6], but actual road data may have only partial or incomplete attribute information, such as OpenStreetMap (OSM) road data [7]. In contrast to attribute information, geometric features are more essential to road networks, so how to identify an overpass using geometric features becomes a problem.

In recent years, an overpass is mainly identified by traditional vector-based methods coming from computational geometry and graph theory. Mackaness and Macchechnie proposed a method to identify and simplify unstructured overpasses [8]. In their method, an overpass was firstly positioned by searching for the denser node areas in road networks, and then the hierarchical relationship between the sections of local road network was constructed. Finally, the structure of the overpass was simplified by a graph simplification method. However, since the internal structure of an overpass is too complex to be accurately identified by their method, which can generate many wrong simplified results, it is necessary to manually judge whether the results are correct. In 2011, Xu proposed an overpass identification method based on structural patterns [9]. In his method, the structure of the road intersection was described using a directional attribute relationship graph, the template library of a typical structure was established and correspond intersections were identified using a relationship graph described by the template. Compared with the method proposed by Mackaness and Macchechnie, the accuracy of the identification results was improved. However, this method relies too much on artificially constructed structural templates and is hard to apply to atypical overpasses. In 2013, Qian proposed an improved structural pattern recognition method [10]. In his method, road topological features were used to describe an overpass, a typical quantitative expression template library was established, and the overpass was identified by comparing the structure with quantitative expressions. In 2016, Ma proposed a method to enrich the structure description in his previous papers [11]. An overpass feature space was established by using six parameters, including length, area, compactness, parallelism, symmetry, and semantic attributes. Then a support vector machine (SVM) was trained and used to identify the main roads and auxiliary roads in each overpass. However, his method cannot identify the overall structure of an overpass directly.

The above methods belong to the identification approach based on artificially designed features. The overpass structure is mapped to a feature description space through spatial calculation, and then the overpasses are identified and classified. Such methods rely too much on the design of the feature items (such as length, angle of intersection, curvature, graph template, etc.). These artificially designed shallow features only work well for standard overpass structures in an ideal state. However, in actual data, an overpass is often under an unsatisfactory state, and there are many disturbing factors. In order to identify overpasses in actual data more accurately, Qian and He proposed to solve the overpass classification problem by a machine learning method in 2018 [12]. In their method, a convolutional neural network (CNN) was trained with images containing overpasses to classify them into different types. Artificially designed feature items were not needed, instead, road data was converted into raster data as the training data for the CNN model. Their method combines vector data and raster images, and uses the neural network learn the fuzzy characteristics of overpasses, and classifies the complex overpass structures in OSM. As far as we know, this method is the first attempt to classify overpasses using a CNN model. In the classification process, feature items are not needed to be given manually, and the uncertainty of the result is reduced by the self-learning mechanism. However, his method can only be used to classified the types of overpasses and cannot be used to identify the location of overpasses in a large range of road networks.

In order to handle the location problem mentioned above, this paper aims to identify overpasses in road networks using a target detection model of deep learning (Faster-Region Convolutional neural network, Faster-RCNN). As there is no labelled data for such problem at present, an overpass labelling geodatabase (OLGDB) for the OSM road network data of six typical cities in China was established. The data flow chart is shown in Figure 1.

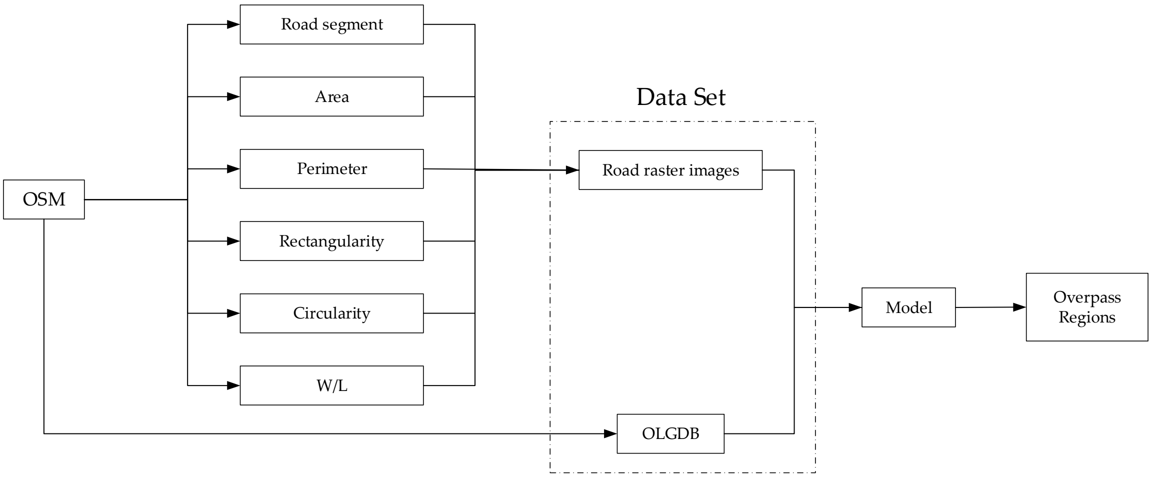


Figure 1. Data flow chart.

In the training data generating phase, we integrated five geometric metrics of roads into image bands and further compared the contribution of these metrics. Finally, three geometric metrics having better effects were selected to be synthetized into the RGB band values in order to use pre-trained CNN models. Three different CNNs (ZF-net, VGG-16, Inception-ResNet V2) were integrated into the Faster-RCNN model and evaluated to find the optimal one by accuracy performance. The results of different model parameters (such as learning rate, batchsize) were measured by fine-turning experiments. The experimental results showed that Faster-RCNN can be applied to the identification of overpasses and good results were achieved. This work can be used to evaluate and supervise the development of urban road networks which is significant in urban science.

2. Research Method

In the field of computer vision, image classification and detection are the focus for research [13,14]. An image classification model is used to divide each image into a single category, usually corresponding to the most prominent object in the image [15–17]; but in the real world, in general, images do not only contain one object, so it is not accurate to simply assign images to a single label. For such a case, a target detection model is needed and used to identify multiple objects in an image and to determine the position of objects in images using rectangular bounding boxes [18–20]. So, in this paper, Faster-RCNN, which meets the above requirement, is studied to identify overpasses in road networks.

The research route is as follows, firstly, OSM road network data is preprocessed (mainly including data screening, data labelling, calculation of geometric metrics, data enhancement and conversion) in order to generate the training data needed by the model. Then, training data is used to train Faster-RCNN and the model parameters and structure are optimized through contrast experiments. Finally, the optimal model is used to detect road networks and overpasses are marked using a rectangular bounding box. The technical flow chart is shown in Figure 2.

2.1. Data Preprocessing

The data source for this paper is OSM road data. OSM is an online map collaboration designed to create a free world map that everyone can edit [21,22]. So OSM data is a typical kind of volunteered geographic information (VGI) data, which can be voluntarily contributed by anyone who wants to: children or adults, amateurs or experts, they may have different motivations and may even be contributing without their knowledge [23,24]. Data contributors use satellite images, GPS devices, and traditional regional maps to ensure accuracy and timeliness.

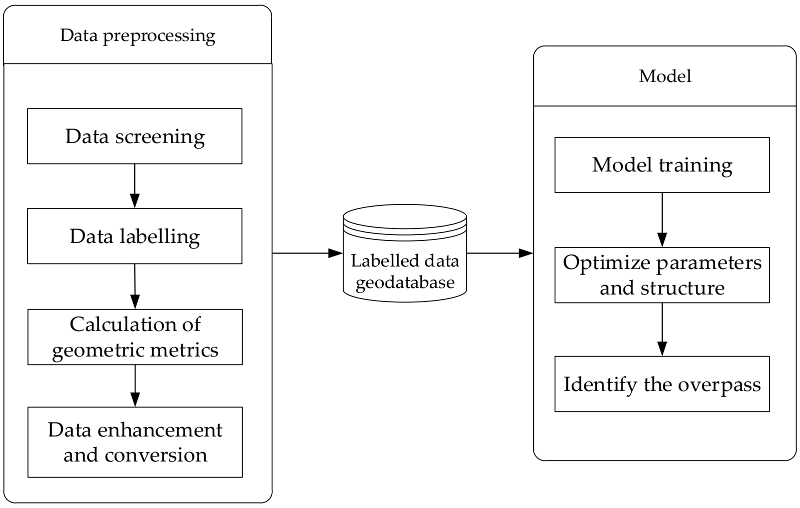

Data preprocessing
Data screening
Data labelling
Calculation of geometric metrics
Data enhancement and conversion
Labelled data geodatabase
Model
Model training
Optimize parameters and structure
Identify the overpass

Figure 2. Technical flow chart.

There are many advantages of OSM, such as short update periods, convenient data acquisition access, and good data quality [25]. These characteristics make OSM data an ideal source for many scientific research experiments. The utility of OSM has been studied as a potential source of road data and the results have been promising [26,27]. Thus, OSM road data is selected as the data source for this paper.

2.1.1. Data Screening

In OSM data, road data has a type attribute (Fclass) which represents the category of roads and includes attribute values such as motorway and step [28]. Before an overpass is labelled, road data is filtered according to the Fclass attribute, which aims to eliminate some irrelevant data such as pedestrian bridges and tunnels. Compared with the initial data, filtered road data is more significant, which is beneficial to the study of Faster-RCNN on the focus of overpass structures. Parts of valid Fclass attribute values of OSM road data are shown in Table 1:

Table 1. Attribute values for the Fclass attribute.

Attribute Value	Comment
Motorway	A restricted access major divided highway, normally with two or more running lanes plus an emergency hard shoulder.
Secondary	The next most important road in a country's system, often linking towns.
Steps	For flights of steps on footpaths.
Primary	The next most important road in a country's system, often linking larger towns.
Trunk_link	The link road leading from a trunk road to another trunk road or a lower class highway.
Path	A non-specific path.

2.1.2. Data Labelling

In this paper, an OLGDB is established to store all the labelled overpasses which were produced by ArcGIS software on a OSM road network and confirmed by Google Maps. Overpass structures are labelled with vector polygons rather than rectangular bounding boxes (a traditional target detection task is to select an object in images as a positive sample using a rectangular bounding box). This facilitated subsequent data enhancement operations:

1. Avoiding an invalid margin while using pure image enhancement;
2. Vector rotation and polygon labelling could make the minimum enclosing rectangle (MER) recalculate correctly.

The illustration of the above two reasons is shown in Figure 3.

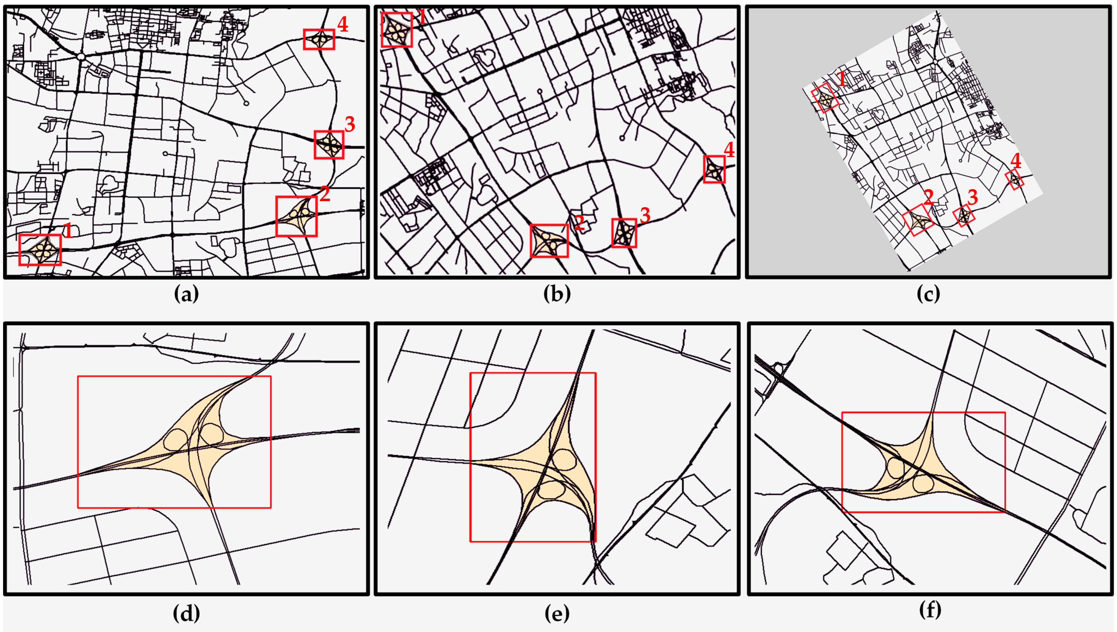


Figure 3. Generated training samples: (**a**) without any rotation; (**b**) with 125° vector rotation; (**c**) invalid margin appears with 75° raster rotation; (**d**) an amplified overpass and its minimum enclosing rectangle (MER); (**e**) the same overpass with 120° rotation and its novel MER; (**f**) the same overpass with 150° rotation and its novel MER.

There are 640 labelled overpasses in the OLGDB, part of them are shown in Figure 4.

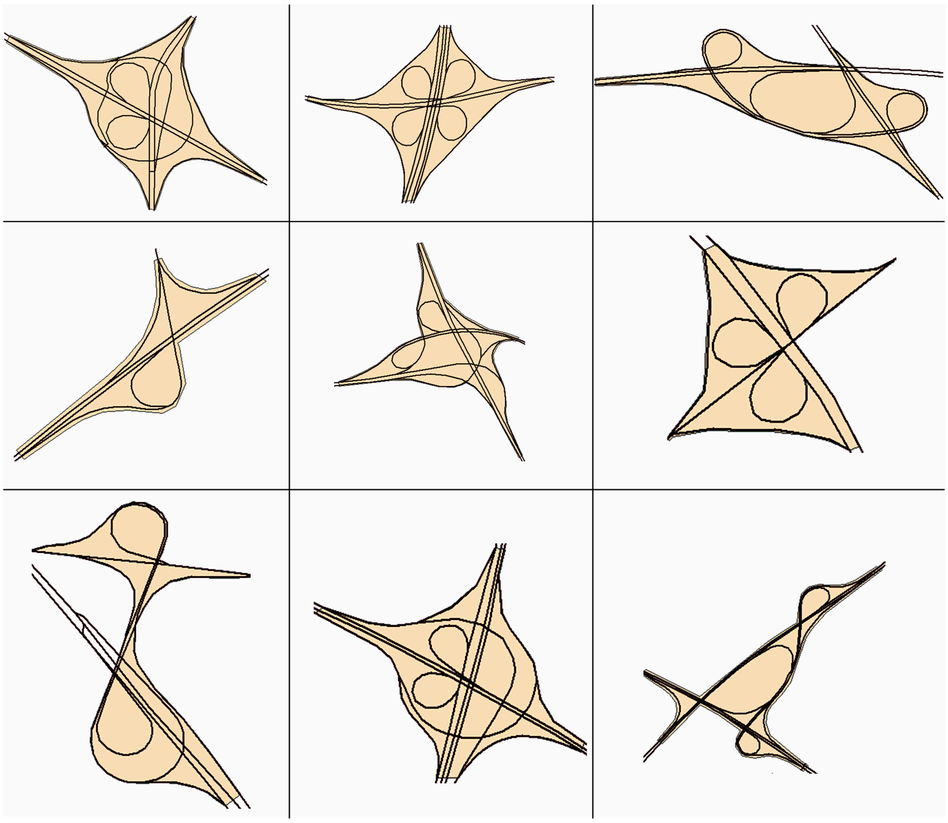


Figure 4. The labelled overpass data.

2.1.3. Calculation of Geometric Metrics

In the generation of training data, geometric information is only partially kept with the conversion from vector to raster. So geometric metrics which measure morphology diversities in field form need to be included to enhance the presentation ability. Five different geometric metrics, including perimeter, area, squareness, circularity, and W/L (the width to length ratio of the circumscribed rectangle), are calculated. Three of them with better effects are selected as the RGB band values of training data through experiment comparison, corresponding to the 3D input variant of pre-trained CNN structures in Faster-RCNN.

The polyline-based network skeleton model is converted to a polygon-based street block model by ArcGIS in order to get the five geometric metrics. The perimeter and area of the block surrounded by road sections can be obtained directly.

Rectangularity is the ratio of block area to the area of the MER, reflecting the filling degree of a block to its MER. The formula is as shown in Equation (1):

$$\mathrm{R} = A_o / A_{MER} \tag{1}$$

where R is the rectangularity of a block, A_O is the block area and A_{MER} is the MER area of a block. R is between 0 and 1. The more curved the block, the smaller the rectangularity.

Circularity is an important geometric metric in image processing and it can be used to describe the closeness of a polygon to a circle. The circularity of a circle is 1. The smaller the circularity, the more irregular the image and the larger the gap with circle. The calculation formula of circularity is as shown in Equation (2):

$$\mathrm{E} = (4\pi * A_O) / (P * P) \tag{2}$$

where A_O is the block area and P is the block perimeter.

W/L is the width to length ratio of the MER. W/L is used to distinguish an elongated block from a round or square block. The calculation is shown in Equation (3):

$$\mathrm{W/L} = W_{MER} / L_{MER} \tag{3}$$

where W_{MER} represents the width of the MER and L_{MER} represents the length of the MER.

2.1.4. Data Enhancement

Deep learning models require a large amount of training data to ensure accuracy, but the overpasses in road networks are relatively sparse, so two kinds of data enhancement are employed to amplify the training data. The first type of data enhancement is to enhance the overpass and its description polygon by vector rotation conversion. After the overpass vector data is rotated, the relative coordinates of positive samples will change. The coordinate variation formula of rotation conversion is shown in Equations (4) and (5):

$$X_{new} = x * \cos\alpha + y * \sin\alpha \tag{4}$$

$$Y_{new} = y * \cos\alpha - x * \sin\alpha \tag{5}$$

where X_{new} and Y_{new} are converted coordinates and α is the angle at which vector data rotates clockwise.

The second type of data enhancement is to enhance the converted raster data. There are eight patterns for an image, as shown in Figure 5.

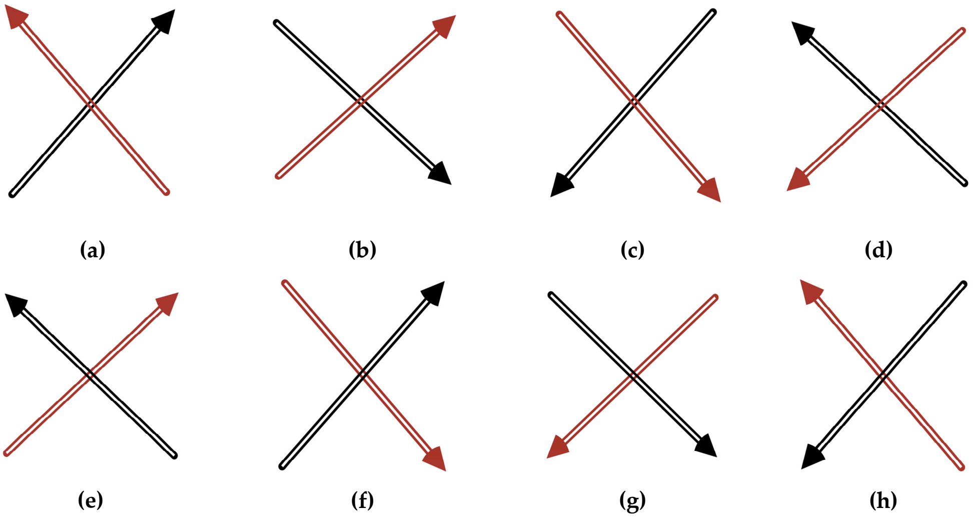


Figure 5. The eight rotate and flip patterns for an image. (**a**) the initial state; (**b**) rotate 90 degrees clockwise; (**c**) rotate 180 degrees clockwise; (**d**) rotate 270 degrees clockwise; (**e**) right left flip; (**f**) right left flip and rotate 90 degrees clockwise; (**g**) right left flip and rotate 180 degrees clockwise; (**h**) right left flip and rotate 270 degrees clockwise.

Therefore, we rotate and flip each training image to generate eight images with different axis change rules. When an image undergoes the above-described rotation and transposition, the size of the image remains unchanged, the information in the image is not lost. This not only ensures the accuracy of the training data, but also increases the sample number of overpasses. After the image data is enhanced, the relative coordinates of overpass structure in each training image are obtained by coordinate conversion formula. Coordinate conversion formulas are as shown in Equations (6)–(13) (taking the form of Figure 3b,e as examples):

$$X_{bmin} = Y_{amin} \tag{6}$$

$$Y_{bmin} = width - X_{amax} \tag{7}$$

$$X_{bmax} = Y_{amax} \tag{8}$$

$$Y_{bmax} = width - X_{amin} \tag{9}$$

$$X_{emin} = width - X_{amin} \tag{10}$$

$$Y_{emin} = Y_{amin} \tag{11}$$

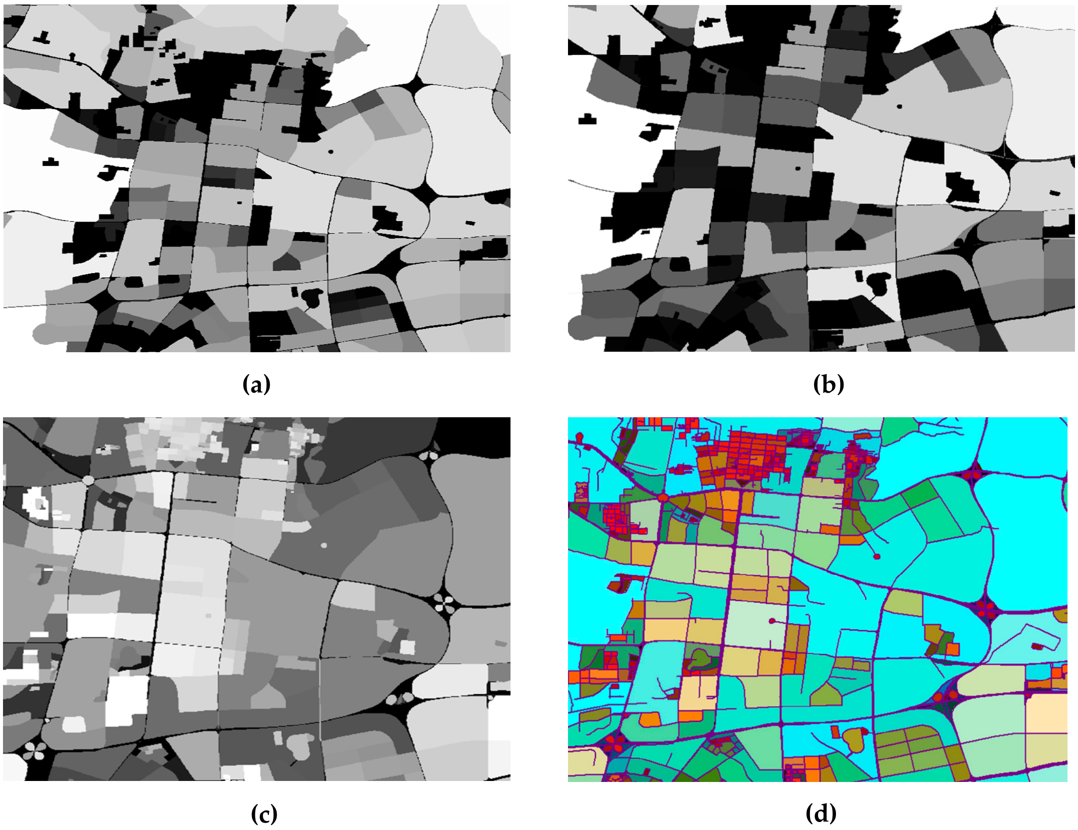
$$X_{emax} = width - X_{amax} \tag{12}$$

$$Y_{emax} = Y_{amax} \tag{13}$$

The boundary coordinates of the positive samples in the training image of Figure 3b are obtained by Equations (6)–(9), X_{bmin}, Y_{bmin}, X_{bmax}, Y_{bmax} represent the upper-left and lower-right coordinates of the positive sample region in Figure 3b, X_{amin}, Y_{amin}, X_{amax}, Y_{amax} represent the upper-left and lower-right coordinates of the positive sample region in the initial configuration; the boundary coordinates of the overpasses in the training image of Figure 3e are obtained by Equations (10)–(13). Width represents the width of training image. After the above data enhancement, the problem that the training data is rare can be solved as 12,000 training images can be provided to the learning model.

2.1.5. Data Conversion

After three geometric metrics with better effects are selected (as described in Section 2.1.3), the values of the geometric metrics are adjusted to 0–255 space, and then, the data conversion function of the Geospatial Data Abstraction Library (GDAL) is used to convert vector road data to raster and to keep adjusted geometric metrics in RGB bands of raster images. GDAL is an open source raster spatial data conversion library under the X/MIT license, which uses an abstract data model to access various geodata file formats and also has a series of command-line tools for data conversion and processing. The converted raster images are shown in the Figure 6.


(a)

(b)

(c)

(d)

Figure 6. (**a**–**c**) the raster image with one geometric metric; (**d**) the raster image with three geometric metrics as RGB bands.

2.2. *Target Detection Model*

This paper aims to identify overpasses in road networks using a target detection model of deep learning (Faster-RCNN). In Faster-RCNN, a convolutional neural network (CNN) is used to extract road features and generate feature maps; the backend contains RPN and ROI pooling; it can generate, adjust, and classify regions.

2.2.1. Convolutional Neural Networks

A convolutional neural network (CNN) is a feedforward neural network and its artificial neurons can respond to a part of surrounding areas. It has excellent performance for large image processing [29]. A CNN consists of one or more convolutional layers, fully connected layers, and pooling layers. This structure enables the convolutional neural network to take advantage of the two-dimensional structure of input data. CNNs give better results in image and speech recognition tasks than other deep learning structures, and can also be trained using backpropagation algorithms [30]. Compared with other depth and feedforward neural networks, CNNs require fewer parameters, making it an attractive deep learning substructure.

CNN provides an end-to-end learning model for image feature extraction and learning. The parameters in the model can be trained by traditional gradient descent methods. Trained CNN can learn features in image and can extract and classify those features. As an important research branch in the field of neural networks, CNN is characterized by the features of each layer being calculated by the local region of the upper layer through the sharing of the convolution kernel of weight. This characteristic makes CNN more suitable for learning and expressing image features than any other neural network architectures.

In a CNN, the convolution calculation is performed by a sliding mechanism and convolution kernel. The elements of a convolution kernel matrix and the elements of a covered area are multiplied and accumulated. Each convolution kernel is a feature extractor, and the convolution operations at all locations in training images share the weight of convolution kernel. In general, in a target detection model, a single convolution kernel is not sufficient, and multiple convolution kernels are needed to extract different levels of features in training images. In a CNN, the features of training images are stored in the form of feature maps. Assuming that the *j*-th feature map of the *l*-th layer is X_i^l, then the convolution calculation of X_i^l is as shown in Equation (14).

$$X_i^l = f\left(\sum\nolimits_{i \in M_j} X_i^{l-1} * k_{ij}^l + b_j^l\right) \tag{14}$$

where k_{ij}^l is the weight parameter of convolutional layer, b_j^l is the bias variable parameter, and f is the activation function, which is used to apply nonlinear transformation on input values.

In order to reduce the size of parameters, there can be a pooling layer after convolutional layer. The role of the pooling layer is to reduce the size of the feature map. Assumed that the feature map after pooling is X_j^{l+1}, the calculation formula of pooling operation is as shown in Equation (15).

$$X_j^{l+1} = f\left(\beta_j^{l+1} down\left(X_j^l\right) + b_j^{l+1}\right) \tag{15}$$

where β_j^{l+1} is the weight parameter of the pooling layer, b_j^{l+1} is the bias variable parameter and *down* is the pooling operation types. For example, when the size of the pooling layer is set to be 2 × 2 and the type of pooling operation is "MAX", it means that the pixel value in the range of 2 × 2 is replaced by its maximum value. f is the activation function.

There are a variety of CNNs, such as the classic LeNet-5 model, which are mainly used in some fundamental computer vision applications, such as handwritten character recognition and image classification. The Convolutional Deep Belief Network that is generated from a CNN and Deep Belief Network has been successfully applied to face feature extraction as an unsupervised generation model [31–33]. A Full-convolution network (FCN) realizes end-to-end image semantic segmentation and greatly surpasses traditional semantic segmentation algorithm in accuracy [34]. Alex Net has achieved breakthroughs in the field of massive image classification [35]. RCNN based on region feature extractions has achieved success in the field of target detection [36]. The target detection model used in this paper is the Faster-RCNN model under the RCNN series model.

2.2.2. Faster-RCNN Model

The Faster-RCNN model is a deep learning target detection model [37]. It uses CNN as a substructure and greatly improves the effect of target detection. It is a two-stage detection algorithm. This algorithm divides the detection problem into two phases:

1. Generating candidate regions.
2. Classifying candidate regions.

The structure of Faster-RCNN model is shown in Figure 7.

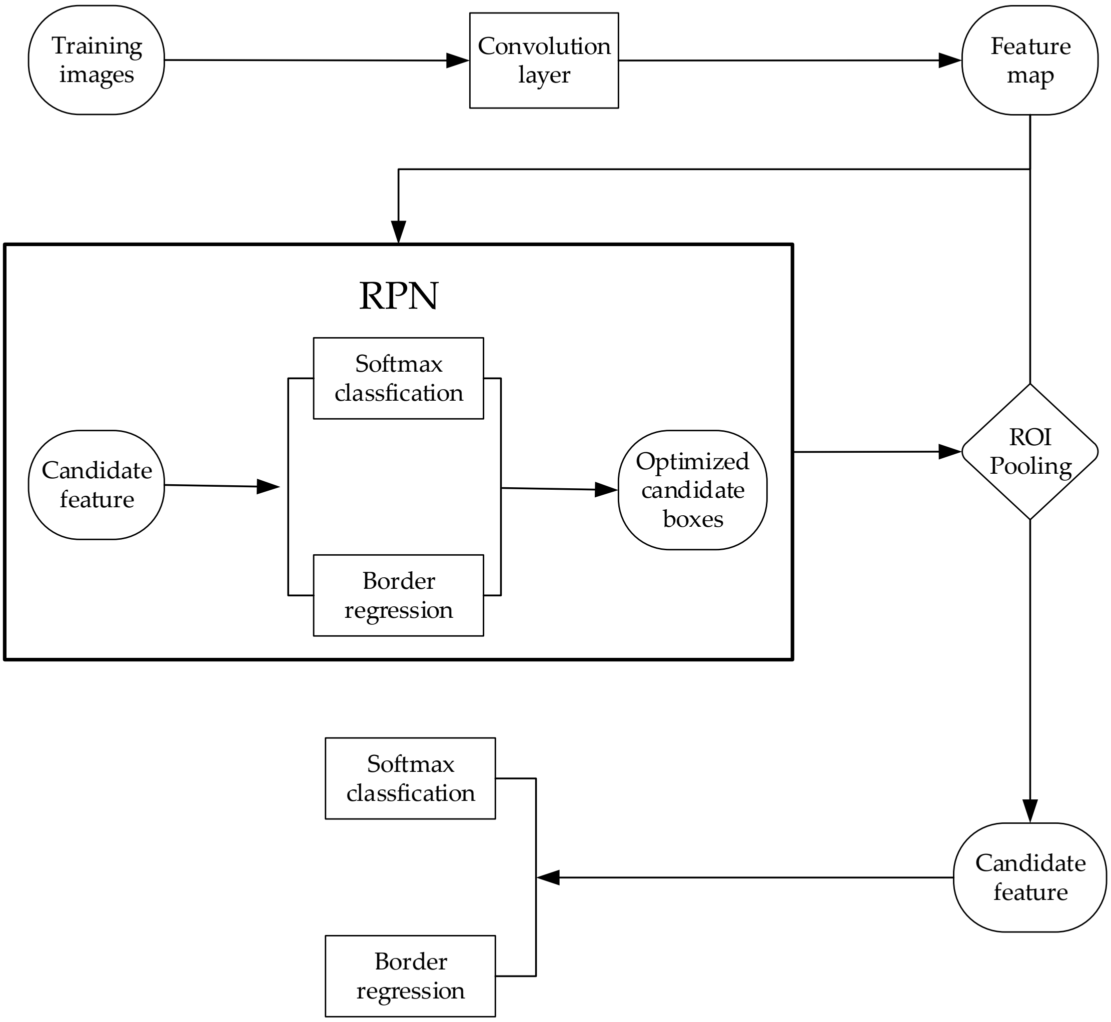


Figure 7. The structure of the Faster-Region Convolutional neural network (Faster-RCNN) model.

The Faster-RCNN model is an improvement of the Fast-RCNN model. In the Fast-RCNN model, the generation of candidate regions is realized by a selective search algorithm, which is a very time-consuming process. The Faster-RCNN model is improved on this point and the Region Proposal network is introduced to generate candidate regions, namely Faster-RCNN = Fast-RCNN + RPN [38]. In the Faster-RCNN model, the feature map of an image is firstly generated by CNN, and then the generated feature map is input into the RPN network. The RPN network obtains candidate regions and enters them into the ROI pooling layer. The ROI pooling layer receives both the optimized candidate boxes of RPN output and feature map of convolution layer output, then the regions in the feature maps corresponding to the optimized candidate boxes are pooled.

The corresponding features of candidate regions are extracted in feature maps. Finally, two fully connected layers are entered, one is the classification layer (Softmax classification), which is a multi-objective classification layer, giving each candidate region a category label, and the other is the coordinate regression layer, which can be used to adjust the coordinates of the candidate areas.

In the RPN network, features are extracted from feature maps using a sliding window, and each sliding window generates k candidate regions with different sizes and different aspect ratios (generally using 3 different sizes and 3 different aspect ratios, i.e., k = 9). Next, the corresponding feature of each candidate region is input to two fully connected layers. One is a classification layer (Softmax classification). This classification layer performs binary predictions to determine whether each candidate region is an object. The other is the coordinate regression layer, which is used to adjust the coordinates of candidate regions. The RPN network shares the same group of convolutional layers with Fast-RCNN. This strategy can simultaneously train and adjust the RPN network and Fast-RCNN. Therefore, the Faster-RCNN model becomes a completed end-to-end model, and the training speed and detection accuracy of the model are greatly improved.

For training RPN, if a candidate region is one of the following two cases, the label of the candidate region is positive.

1. the candidate region with the highest Intersection-over-Union (IOU) overlap with a ground-truth box.
2. a candidate region that has an IOU overlap higher than 0.7 with any ground-truth box.

Note that a single ground-truth box may assign positive labels to multiple candidate regions. Usually the second condition is sufficient to determine the positive samples; but the first condition must be considered for the reason that, in some rare cases, the second condition may find no positive sample. The label of a candidate region is negative if its IOU is lower than 0.3 for all ground-truth boxes. Candidate regions that are neither positive nor negative do not contribute to the training objective.

In the Faster-RCNN model, the loss function is defined as shown in Equation (16).

$$\mathrm{L}(\{p_i\}\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}\left(p_i, p_i^*\right) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}\left(t_i, t_i^*\right) \tag{16}$$

The loss function has two parts. The classification loss L_{cls} is log loss over two classes (object vs. not object), i is the index of a candidate region, p_i is the predicted probability of region i being an object. The ground-truth label p_i^* is 1 if the candidate region is positive, and is 0 if the candidate region is negative. The other one is the regression loss, t_i is a vector representing the four parameterized coordinates of the predicted bounding box, and t_i^* is that of the ground-truth box associated with a positive region. For boundary box regression, we use Equations (17)–(20) to calculate the coordinates.

$$t_x = (x - x_a)/w_a,\ t_y = (y - y_a)/h_a \tag{17}$$

$$t_w = \log(w/w_a),\ t_h = \log(h/h_a) \tag{18}$$

$$t_x^* = (x^* - x_a)/w_a,\ t_y^* = (y^* - y_a)/h_a \tag{19}$$

$$t_w^* = \log(w^*/w_a),\ t_h^* = \log(h^*/h_a) \tag{20}$$

where x, y, w, and h denote the box's center coordinates and its width and height. Variables x, x_a, and x^* are for the predicted box, region box, and ground truth box, respectively (likewise for y, w, h).

3. Experiment and Result Analysis

In this paper, Beijing, Shanghai, Shenzhen, Wuhan, Zhengzhou, and Lanzhou in China were selected as the experimental research areas. Beijing, Shanghai and Shenzhen were selected as representatives of first-tier cities. Beijing is the capital of China and its urban roads are all in a chessboard pattern. By the end of 2018, the city's highway mileage was 22,255.8 km. Shanghai as a municipality directly under the Central Government of China is located in the Yangtze River Delta with a highway mileage of over 16,000 km. Shenzhen is one of the four first-tier cities in China. It is located in the southern Guangdong Province and is a national transportation hub city. Wuhan, Zhengzhou, and Lanzhou were selected as representatives of second-tier cities. Wuhan is located in the central part of China and the Jianghan Plain in eastern Hubei Province. The highway mileage exceeds 15,000 km. Zhengzhou is an important central city in central China, with a total main highway length of approximately 12,700 km. Lanzhou is one of the important central cities in western China and a transportation hub in the northwest. These cities, as representatives of China's first- and second-tier cities, are located in different parts of China, which makes research data more representative and diverse. The road data for the selected cities are shown in Figure 8.

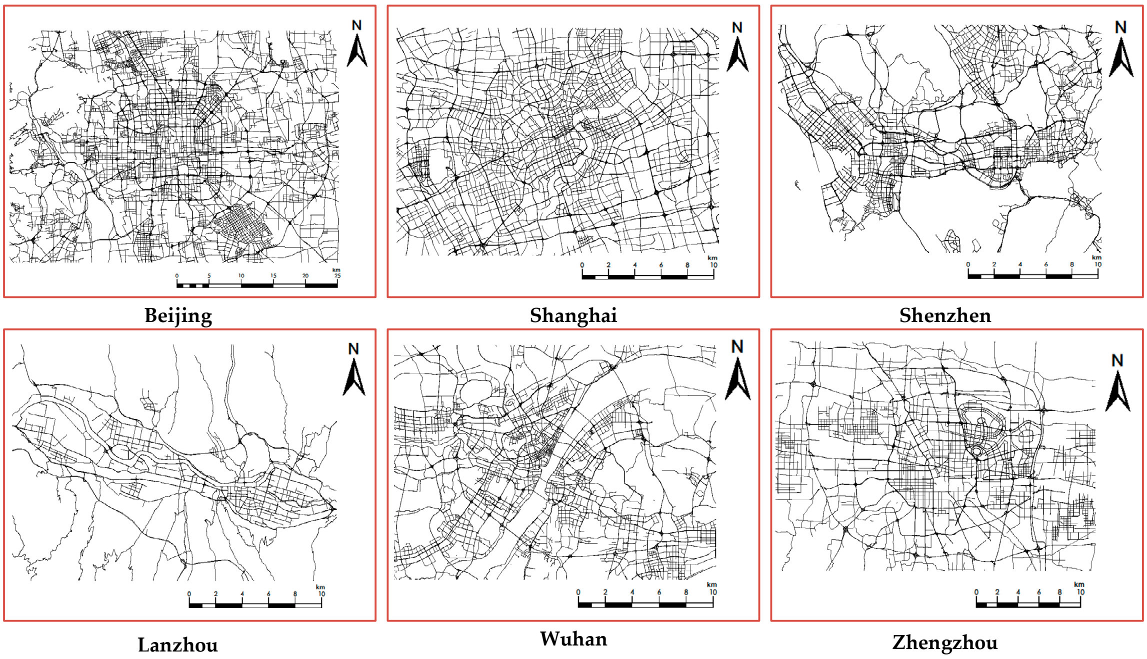


Figure 8. The road data for the selected cities.

In this paper, a series of comparative experiments are designed and conducted to study the influence of different geometric metrics, model structures, and model parameters on the overpass identification task. Experiment 1 tests the distinct effects of CNN substructure. We integrated three different CNNs (ZF-net, VGG-16, Inception-ResNet V2) into Faster-RCNN. These convolution structures are normally used to accomplish feature extractions. In addition, they have better performance in feature extraction for complex structures because of their sophisticated layer design. The three CNNs are shown in Figure 9.

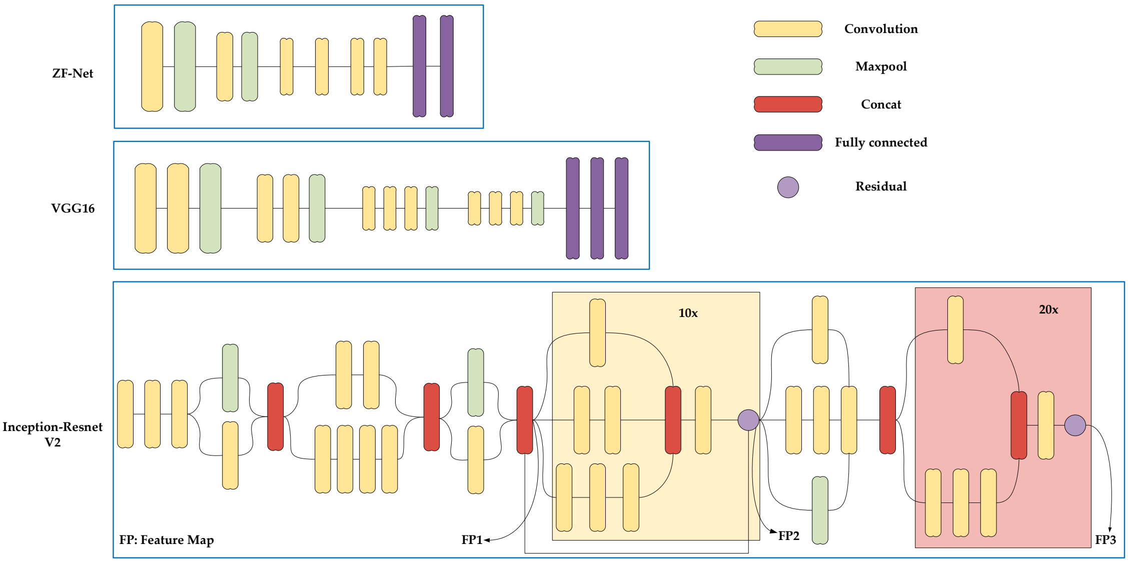


Figure 9. The three convolutional neural network (CNN) substructures.

ZF-net consists of five convolution layers and two fully connected layers [39]; VGG-16 has 16 total convolutions and fully connected layers [40]; Inception-ResNet V2 introduces two special modules: Inception and residual [41]. In inception modules, multiple convolution branches are linked to a single feature map, while residual modules (those that repeat 10 or 20 times in Inception ResNet V2 in Figure 6) add input feature maps to the convoluted output.

In order to compare the performance of the three CNNs, each model trained 20,000 iterations under the same model parameters and the performance of each model in identifying overpasses under the same validation data set was evaluated. The results are shown in Figure 10.

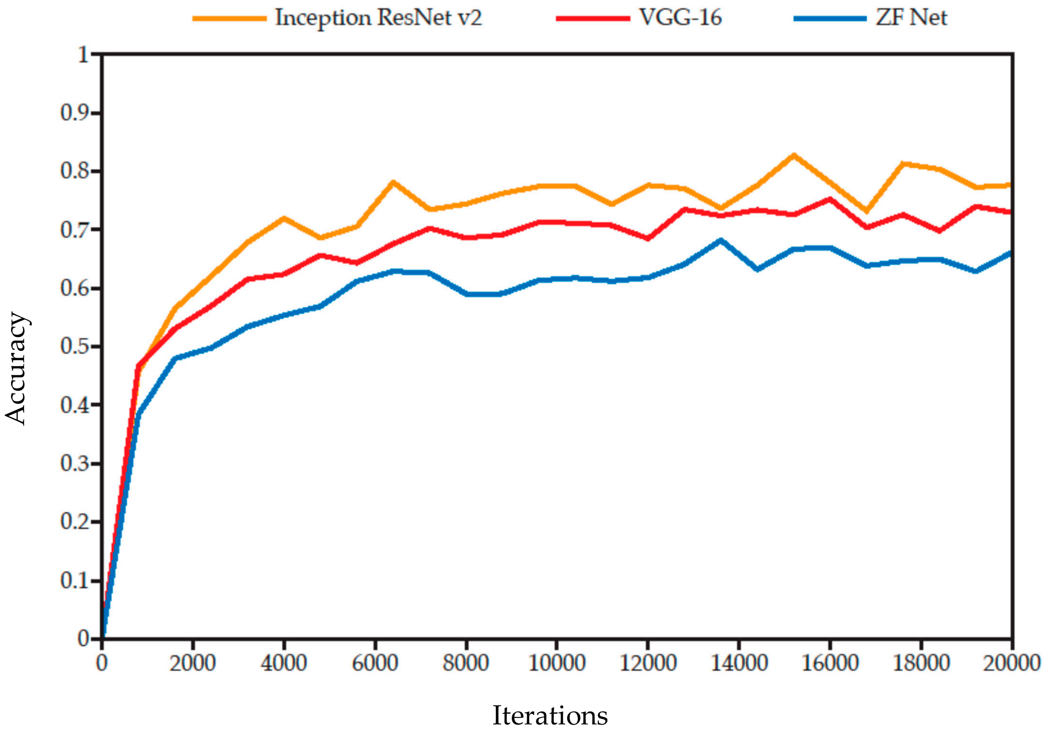


Figure 10. The validation accuracy of three CNN substructures.

The results showed that VGG-16 and Inception-ResNet V2 have better network performance than ZF-Net because they have a deeper network structure. The model with integrated Inception-ResNet V2 convolution structure had the highest prediction accuracy, because Inception-ResNet V2 combines the advantages of a residual connection and an inception module. Residual connections solve the problem of gradient propagation in deep connected networks, and even if the network structure is deep, it can be easily learned. Inception modules allow network extending without increasing computational cost. At each level, the module applies convolution filters of different sizes in parallel on the same input map and their results are connected to the same output. This strategy integrates multi-scale information to ensure better performance. Since we take identification accuracy as the primary goal of the experiment, Inception-ResNet V2 is used as the CNN of the final Faster-RCNN model.

Experiment 2 tests the effects of different model parameters on the detection of overpasses. In Faster-RCNN, learning rate and batchsize are the two parameters that have the greatest impact on model results. Learning rate is a parameter that guides the model in adjusting the weight of the network through a loss function. The lower the learning rate, the slower the change in loss function. Batchsize is the number of samples used in one iteration. In a CNN, increasing the value of batchsize usually makes network converge faster, but due to the limitations of memory resources, if batchsize is too large, memory shortage will be insufficient, and the time consumed by the model to reach same accuracy will increase. Therefore, 12 sets of comparative experiments were performed using three different learning rates (0.0001, 0.0005, and 0.001) and four different batch sizes (5, 10, 15, and 20). The results of comparison experiment are shown in Figure 11:

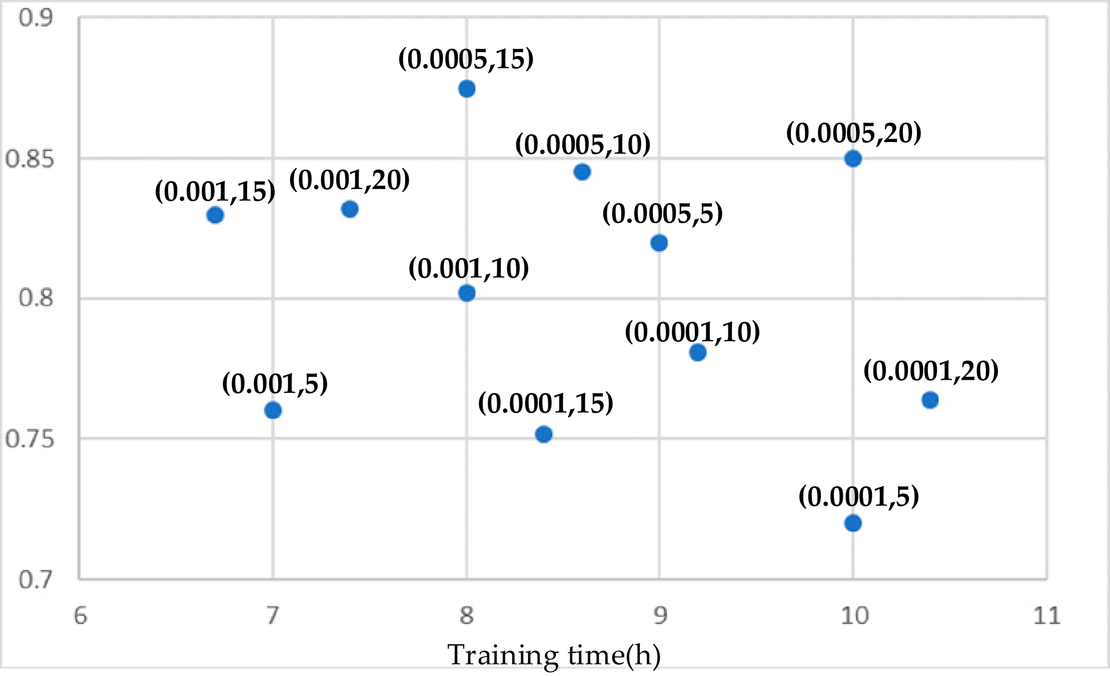


Figure 11. The validation accuracy and training time under different model parameters.

The results showed that when learning rate is 0.0005 and batchsize is 15, the model gives the best identification result. When the value of batchsize increases, the accuracy of the model does not changed substantially, and it takes too long to reach the same precision.

After training data was selected according to the method discussed in Section 2.1, Experiment 3 compared the detection effect of training data with different geometric metrics. The results reflected the contribution of different geometric metrics to the identification of overpasses. The experiment results are shown in Table 2.

Table 2. The results of different geometric metrics.

Geometric Metric	Accuracy
perimeter	0.79
area	0.824
rectangularity	0.887
circularity	0.474
W/L	0.682
perimeter and area and rectangularity	0.902

The results showed that three geometric metrics (area, perimeter, rectangularity) have better performance on overpass identification, therefore, these adjusted geometric metrics are selected to synthetize the values of RGB bands in converted raster data.

After the above comparative experiments, the optimized Faster-RCNN model was established and used to identify overpasses. The identification accuracy and the loss are shown in Figure 12.

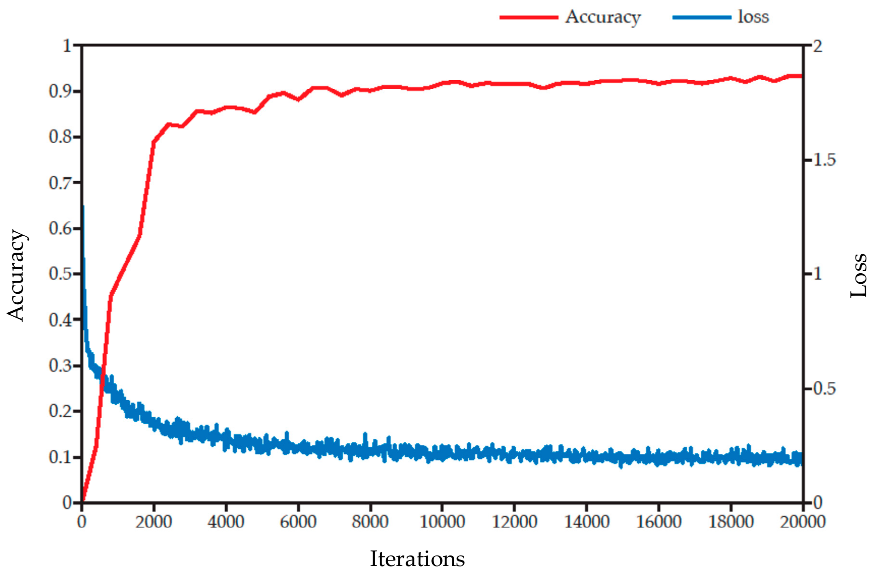


Figure 12. The training accuracy and the loss of the final model.

The identification results are shown in Figure 13.

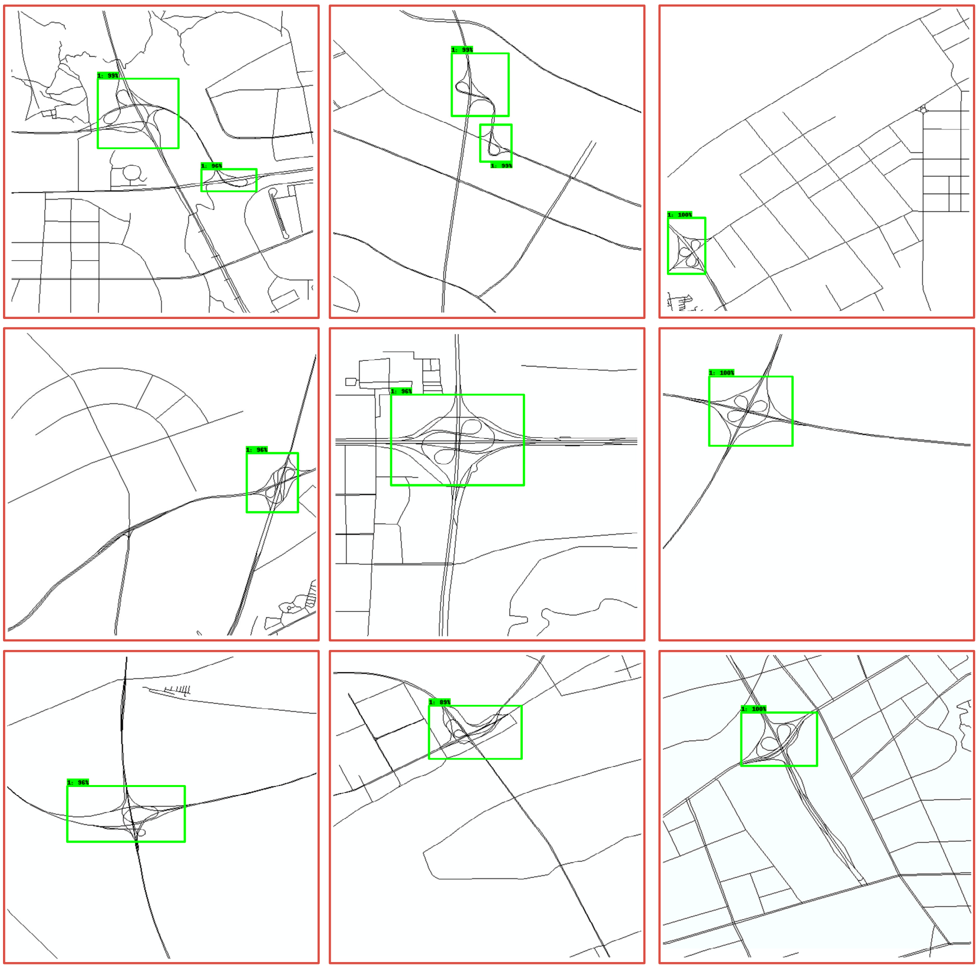


Figure 13. Identification results of overpasses. The range selected in the green box is identified by the model, the value is the probability that the model thinks the range chosen by the green box is the truth.

4. Conclusions

In this paper, Faster-RCNN was used to extract and learn the features of overpass structures in road networks and applied to overpass identification. The OLGDB was established to store labelled overpass data. The optimal model was obtained by evaluating the performances of various models with different geometric metrics-based training data, CNN structures, and training parameters. The accuracy of overpass identification reached up to more than 90%. The results showed that Faster-RCNN is effective for overpass identification. Faster-RCNN can learn the features of road networks well and can determine the position of overpasses in a complex road network with high accuracy. To the best of our knowledge, this study is the first attempt to identify overpasses using Faster-RCNN. Compared with traditional vector methods, this method does not require artificially designed features and avoids the uncertainty of experimental results caused by human intervention. The method performs well on actual road network data. In the future, we will expand the OLGDB, switch to more sophisticated target detection models, and attempt to identify other road network patterns under the deep learning framework. Also, we will continue to explore the applicability of this method in data of different scales or different urban road networks. And, besides OSM road data, we will discuss the impact of data from different sources on the identification results, such as road networks with more interfered road sections or road networks with missing parts of road sections.

Author Contributions: Maosheng Hu and Hao Li proposed the idea and wrote the paper, Hao Li built the model and performed the experiments; Youxin Huang analyzed the data.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No. 41501433).

Acknowledgments: We really appreciate the two anonymous reviewers for their useful comments and suggestions. This study was supported by the National Natural Science Foundation of China (Grant No. 41501433).

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

References

1. Studies, J.C.F.U. *The Image of the City*; MIT Press: Cambridge, MA, USA, 1960.
2. Yang, B.; Luan, X.; Li, Q. An adaptive method for identifying the spatial patterns in road networks. *Comput. Environ. Urban Syst.* **2010**, *34*, 40–48. [CrossRef]
3. Heinzle, F.; Anders, K.H.; Sester, M. Automatic detection of pattern in road networks-methods and evaluation. In Proceedings of the Joint Workshop Visualization and Exploration of Geospatial Data, Stuttgart, Germany, 27–29 June 2007.
4. Heinzle, F.; Anders, K.H.; Sester, M. Pattern Recognition in Road Networks on the Example of Circular Road Detection. In Proceedings of the International Conference on Geographic Information Science, Münster, Germany, 23–26 September 2006.
5. Mackaness, W.; Edwards, G. The importance of modelling pattern and structure in automated map generalisation. In Proceedings of the Joint ISPRS/ICA Workshop on Multi-Scale Representations of Spatial Data, Ottawa, ON, Canada, 7–8 July 2002.
6. Scheider, S.; Possin, J. Affordance-based individuation of junctions in Open Street Map. *J. Spat. Inf. Sci.* **2012**, *2012*, 31–56. [CrossRef]
7. Haklay, M.; Weber, P. Openstreetmap: User-generated street maps. *IEEE Pervasive Comput.* **2008**, *7*, 12–18. [CrossRef]
8. Mackaness, W.A.; Mackechnie, G.A. Automating the detection and simplification of junctions in road networks. *GeoInformatica* **1999**, *3*, 185–200. [CrossRef]
9. XUZhu, M.; Yanzi, L. Recognition of structures of typical road junctions based on directed attributed relational graph. *Acta Geod. Cartogr. Sin.* **2011**, *40*, 125.
10. Wang, X.; Qian, H.; Ding, Y.; Zhang, X.; Liu, R. The Integral Identification Method of Cloverleaf Junction Based on Topology and Road Classification. *J. Geomat. Sci. Technol.* **2013**, *30*, 324–328.
11. Ma, C.; Sun, Q.; Chen, H.; Wen, B. Recognition of Road Junctions Based on Road Classification Method. *Geomat. Inf. Sci. Wuhan Univ.* **2016**, *41*, 1232–1237.

12. Haiwei, H.E.; Qian, H.; Xie, L.; Duan, P. Interchange Recognition Method Based on CNN. *Acta Geod. Cartogr. Sin.* **2018**, *47*, 385–395.
13. Chang, C.I.; Ren, H. An experiment-based quantitative and comparative analysis of target detection and image classification algorithms for hyperspectral imagery. *IEEE Trans. Geosci. Remote Sens.* **2000**, *38*, 1044–1063. [CrossRef]
14. Lange, S.; Riedmiller, M. *Evolution of Computer Vision Subsystems in Robot Navigation and Image Classification Tasks*; Springer: Berlin/Heidelberg, Germany, 2004.
15. Majumder, U.K. Machine Learning (ML) Algorithms: An overview of various techniques for target detection and classification (Conference Presentation). In Proceedings of the Society of Photo-optical Instrumentation Engineers, Anaheim, CA, USA, 11–13 April 2017.
16. Haykin, S.; Kosko, B. *GradientBased Learning Applied to Document Recognition*; Wiley-IEEE Press: New York, NY, USA, 2001.
17. Vaillant, R.; Monrocq, C.; Lecun, Y. An Original approach for the localisation of objects in images. In Proceedings of the International Conference on Artificial Neural Networks, Amsterdam, The Netherlands, 13–16 September 1993.
18. Vink, J.P.; Haan, G. Comparison of machine learning techniques for target detection. *Artif. Intell. Rev.* **2015**, *43*, 125–139. [CrossRef]
19. Fu, L.; Fang, S.; Xu, X. Human Motion Target Detection Based on Computer Vision. *Acta Armamentarii* **2005**, *26*, 766–770.
20. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2014.
21. Brinkhoff, T. Open Street Map Data as Source for Built-Up and Urban Areas on Global Scale. *ISPRS Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2016**, *41*, 557–564. [CrossRef]
22. Wang, M.; Li, Q.; Hu, Q.; Zhou, M.; Li, Q.Q.; Hu, Q.W. Quality Analysis on Crowd Sourcing Geographic Data with Open Street Map Data. *Geomat. Inf. Sci. Wuhan Univ.* **2013**, *38*, 1490–1494.
23. Fonte, C.C.; Antoniou, V.; Bastin, L.; Estima, J.; Arsanjani, J.J.; Bayas, J.C.L.; See, L.; Vatseva, R. Assessing VGI data quality. In *Mapping and the Citizen Sensor*; Ubiquity Press: London, UK, 2017; pp. 137–163.
24. Goodchild, M.F. Citizens as sensors: The world of volunteered geography. *GeoJournal* **2007**, *69*, 211–221. [CrossRef]
25. Haklay, M. How good is volunteered geographical information? A?comparative study of OpenStreetMap and Ordnance Survey datasets. *Environ. Plan. B Plan. Des.* **2010**, *37*, 682–703. [CrossRef]
26. Mahabir, R.; Stefanidis, A.; Croitoru, A.; Crooks, A.; Agouris, P. Geo-Information Authoritative and Volunteered Geographical Information in a Developing Country: A Comparative Case Study of Road Datasets in Nairobi, Kenya. *ISPRS Int. J. GeoInf.* **2017**, *6*, 24. [CrossRef]
27. Ludwig, I.; Voss, A.; Krause-Traudes, M. A Comparison of the Street Networks of Navteq and OSM in Germany. In *Advancing Geoinformation Science for A Changing World*; Springer: Berlin/Heidelberg, Germany, 2011.
28. Graf, F.; Kriegel, H.P.; Renz, M.; Schubert, M. MARiO: Multi-Attribute Routing in Open Street Map. In Proceedings of the International Symposium on Spatial and Temporal Databases, Minneapolis, MN, USA, 24–26 August 2011.
29. Cireşan, D.; Meier, U.; Schmidhuber, J. Multi-column Deep Neural Networks for Image Classification. *arXiv* **2012**, arXiv:1202.2745.
30. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2014**, arXiv:1409.1556.
31. Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. *Neural Comput.* **2014**, *18*, 1527–1554. [CrossRef]
32. Lee, H.; Grosse, R.; Ranganath, R.; Ng, A.Y. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In Proceedings of the 26th Annual International Conference on Machine Learning, Montreal, QC, Canada, 14–18 June 2009.
33. Huang, G.B.; Lee, H.; Learned-Miller, E. Learning hierarchical representations for face verification with convolutional deep belief networks. In Proceedings of the 2012 IEEE Conference on Computer Vision & Pattern Recognition, Providence, RI, USA, 16–21 June 2012.

34. 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.
35. 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.
36. Cai, Z.; Fan, Q.; Feris, R.S.; Vasconcelos, N. A Unified Multi-scale Deep Convolutional Neural Network for Fast Object Detection. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016.
37. Deng, L.; Yu, D. Deep Learning: Methods and Applications. *Found. Trends Signal Process.* **2014**, *7*, 197–387. [CrossRef]
38. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015.
39. Li, W.; Hsu, C. Automated terrain feature identification from remote sensing imagery: A deep learning approach. *Int. J. Geogr. Inf. Sci.* **2018**, 1–24. [CrossRef]
40. Alippi, C.; Disabato, S.; Roveri, M. Moving convolutional neural networks to embedded systems: The alexnet and VGG-16 case. In Proceedings of the 17th ACM/IEEE International Conference on Information Processing in Sensor Networks, Porto, Portugal, 11–13 April 2018.
41. Ferreira, C.A.; Melo, T.; Sousa, P.; Meyer, M.I.; Shakibapour, E.; Costa, P.; Campilho, A. Classification of Breast Cancer Histology Images Through Transfer Learning Using a Pre-trained Inception Resnet V2. In Proceedings of the International Conference Image Analysis and Recognition, Póvoa de Varzim, Portugal, 27–29 June 2018; Chapter 86. pp. 763–770. [CrossRef]