Information Sciences 535 (2020) 156–171

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

# Global context based automatic road segmentation via dilated convolutional neural network

Meng Lan [a], Yipeng Zhang [a], Lefei Zhang [a,b,*], Bo Du [a]

[a] National Engineering Research Center for Multimedia Software, Institute of Artificial Intelligence and School of Computer Science, Wuhan University, Wuhan, China
[b] State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan, China

ARTICLE INFO

*Article history:*
Received 2 January 2020
Received in revised form 13 May 2020
Accepted 15 May 2020
Available online 18 May 2020

*Keywords:*
Road segmentation
UNet
Dilated convolution
Global context information

ABSTRACT

Road segmentation from remote sensing images is a critical task in many applications. In recent years, various approaches, particularly deep learning-based methods, have been proposed for accurate road segmentation. However, most existing road segmentation methods always obtain unsatisfactory results (e.g., heterogeneous pixels) due to the complex backgrounds and view occlusions of buildings and trees around a road; consequently, road segmentation remains a challenging problem. In this study, we propose a novel global context based dilated convolutional neural network (GC-DCNN) to address the aforementioned problem. The structure of GC-DCNN is similar to that of UNet. In particular, building the encoder of GC-DCNN with three residual dilated blocks is suggested to further enlarge the effective receptive field and learn additional discriminative features. Thereafter, a pyramid pooling module is used to capture the multiscale global context features and fuse them to achieve stronger feature representation. The decoder network upsamples the fused features to the same size as the input image, combining the high-resolution features with the contracting path of the encoder network. Moreover, the dice coefficient loss is adopted as the loss function. This function differs from those in most previous studies but is more suitable for road segmentation. Extensive experimental results on two benchmark datasets compared with several baseline models demonstrate the superiority of the proposed GC-DCNN algorithm.



## 1. Introduction

Automatic road segmentation from remote sensing images is an important research hotspot in the remote sensing and pattern recognition fields. It plays an essential role in various applications, including vehicle navigation, urban planning, and geographic information system updating [19,35]. However, manually labeling road areas in remote sensing images is extremely time-consuming and tedious. In the past decades, machine learning was applied to various fields, such as images [15,39], natural languages [9] and the Internet of things [10]. Many attempts have been made to use several predefined features with machine learning techniques to realize the automatic road segmentation of remote sensing images, but they have always failed to achieve satisfactory accuracy [37,34]. With the rapid development of deep learning techniques, convolutional neural networks (CNNs) have achieved state-of-the-art performance in many recognition-related tasks, such as image

* Corresponding author.
*E-mail address:* zhanglefei@whu.edu.cn (L. Zhang).

https://doi.org/10.1016/j.ins.2020.05.062
0020-0255/

classification [18,16], object detection [28,4] and semantic segmentation [23,33]. In the remote sensing field, CNNs are also adopted to extract powerful features for various tasks [38,13,7], including road segmentation.

In general, road segmentation labels all road pixels from remote sensing images; hence, this task is actually a binary pixel-level classification task, namely, road pixels and background pixels. Before deep learning techniques were applied to road segmentation, most road segmentation methods were based on pixel-level labeling and road prior. For example, Yuan et al. [37] introduced an oscillator network and adopted a three-step method to gradually achieve the segmentation task. Unsalan et al. [34] proposed a system that contained three complex and interchangeable modules for automatic road extraction. However, these methods frequently predict heterogeneous results and inaccurate road boundaries when dealing with very high-resolution (VHR) remote sensing images with noise and occlusions of trees, cars, and surrounding buildings due to extracted shallow features, excessive human intervention, and inability to cope with a complex background.

Since AlexNet (a simple CNN with five convolutional layers and three fully connected layers) [18] won first place in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 competition, deep learning techniques, as represented by CNNs, have attracted considerable attention in the fields of computer vision and pattern recognition. Subsequently, fully convolutional networks (FCNs) [23] were first proposed for image semantic segmentation, and numerous image segmentation algorithms were developed with remarkable performance improvements, such as encoderdecoder-based models [29,2,27] and the dilated convolution-based DeepLab family [6,5]. These segmentation methods frequently achieve the highest accuracy rates on popular benchmarks, resulting in what many experts regard as a paradigm shift in the field.

With the remarkable progress in the semantic segmentation of natural images, deep CNNs have also been introduced to the road segmentation task of remote sensing images. In [30,1], deep CNNs were applied to extract deep features from remote sensing images. Cheng et al. [7] used an encoderdecoder network with unpooling and deconvolutional operations to improve segmentation performance. However, these plain architectures fail to extract features with rich representation information, which is crucial for remote sensing image analysis. To better utilize spatial contexts, Yang et al. [35] designed a region-based CNN (RCNN) unit to build a deep network with limited memory consumption. This network has a theoretically larger receptive field; however, numerous layers of shared weights in the encoder and the decoder may affect the performance of feature transformation.

Although deep CNN-based methods have progressed considerably in the road segmentation of remote sensing images, their performance can still be improved. Existing deep learning-based methods fail to address the view occlusion problem to obtain coherent and smooth segmentation results because they are simple applications of CNN to the road segmentation task.

To deal with the aforementioned problem, we propose a novel global context-based dilated CNN (GC-DCNN) for road segmentation in the current study. This network can extract efficient features that are combined with multiscale global feature information to enrich final feature representation. In contrast with most previous studies [5,36,21] that always embed the dilated convolution into the last one or two blocks of the feature extractor, such as ResNet-50 and VGG, to enlarge the receptive field while keeping the resolution high, we design a sophisticated residual dilated block (RDB) and use it to build the entire encoder network for feature extraction. With the help of the proposed RDB, the encoder network can handle a wide range of spatial information in each block without increasing the number of parameters, and finally, obtains high-resolution features with a large receptive field and multiscale context information. These features are beneficial for improving feature representation ability and alleviating the effects of noises and occlusions. Before the decoder network restores the extracted features, we use the spatial pyramid pooling module (PPM) to produce multilevel global context features and concatenate them with the original extracted features to form the final features with richer representation information. These features are then upsampled layer by layer by the decoder network. Furthermore, in contrast with most previous methods that use the cross entropy (CE) loss or mean squared error (MSE) loss, we adopt the dice coefficient loss as the loss function in the proposed GC-DCNN. This loss function, which is inspired by medical image segmentation [24], can better consider class imbalance between road pixels and background pixels. The dice coefficient loss is directly optimized on the evaluation metric F1 score, which can better balance accuracy and recall rate in the binary road segmentation task. Extensive experiments have been performed on two public road segmentations of remote sensing image datasets, and the experimental results demonstrate that the proposed GC-DCNN model exhibits state-of-the-art performance compared with other state-of-the-art methods.

The rest of the paper is organized as follows. Section 2 briefly reviews the related literature. Section 3 describes the entire architecture and loss function of GC-DCNN. Section 4 presents the evaluation results and several ablation experiments. Lastly, Section 5 concludes this study.

## 2. Related work

In this section, we review the deep learning based methods and related algorithms for remote sensing image road segmentation.

Since the eight-layer AlexNet [18] won the championship in the ILSVRC Competition 2012, deep learning techniques have been widely studied and applied to various tasks, such as image classification [16,31], object detection [28,8], and semantic segmentation [23]. In the field of semantic segmentation, Long et al. [23] first proposed an FCN for natural scene image segmentation in 2014. This FCN exhibited significant performance improvement compared with traditional segmentation

methods [14,12]. Thereafter, Ronneberger et al. [29] proposed a well-designed symmetric network, called UNet, which used a U-shaped encoderdecoder architecture to deal with the biomedical image segmentation problem. Hierarchical features are extracted from the input images in the encoder network, and then the decoder network restores the final extracted feature maps combined with the corresponding hierarchical features in the encoder network. UNet can predict more precise output and perform better than FCN due to its symmetric structure and feature fusion at different levels of the decoder. The ideas of a symmetric structure and feature fusion have also been introduced in many other fields [20,32,11]. Badrinarayanan et al. [2] and Hyeonwoo et al. [27] preserved the symmetric deep encoderdecoder framework while introducing different structures and operations in an encoderdecoder network. Both studies used the same unpooling operation, in which the indices of the maximum locations selected during pooling operation were recorded and then passed to the decoder part to upsample the feature maps. Another approach for improving segmentation accuracy is to enlarge the receptive field of a network and utilize the global context information. Chen et al. [5] used dilated convolution to obtain a larger receptive field in high resolution and the atrous spatial PPM to capture multiscale global context information.

Several studies pioneered the application of deep learning techniques to the road segmentation of remote sensing images. Zhang et al. [40] proposed the ResUNet framework, which extended UNet with a residual block for facilitating information propagation and achieving improved performance in road segmentation. Cheng et al. [7] developed cascaded deep CNNs for the road segmentation and centerline extraction tasks. The network for road segmentation adopted a symmetric encoderdecoder structure. Mattyus et al. [26] proposed an approach called DeepRoadMapper, in which a CNN-based method was adopted to generate a coarse road segmentation result, and then the binary thresholding and morphological thinning methods were used to construct the final road network graph. However, the DeepRoadMapper works efficiently for images without complex topology and occlusion, which is an ideal case in the real world. In [25], a penalty term was introduced to the binary CE loss to account for topology information. The penalty term was an L2 loss for measuring the difference between high-level features and the ground truth. Furthermore, Bastani et al. [3] proposed the RoadTracer framework, which used an iterative search process guided by a CNN-based decision function and directly obtained a road network graph from the CNN output. The RoadTracer is highly dependent on the performance of the CNN, and thus, may fail to identify some road segments due to errors made by the CNN.

Although these deep learning-based methods have made considerable progress in the field of road segmentation of remote sensing images, they still suffer from several deficiencies in solving the complex background and occlusion problem. First, [7] concluded that road segmentation is actually a binary classification task. Considering the number of training pixels, available computational resources, and expected running time, most deep learning-based methods opt to limit the number of convolutional layers in the entire network to approximately 20, in which the encoder accounts for approximately half, resulting in a extremely limited receptive field on the high-level layers for capturing the long-range context. Second, most existing methods disregard the technique of capturing multiscale context information and fusing it with the extracted high-level features to generate strong representative features; this technique has been utilized in the natural image segmentation field [22,41]. In the current study, we propose GC-DCNN to address the aforementioned problems. GC-DCNN adopts the encoderdecoder structure. The encoder network of GC-DCNN built by dilated residual blocks can access a large range of pixels (spatial information) with minimal discriminative information loss in each block, and finally, produces high-resolution features with a large effective receptive field. With the use of global pyramid processing, multiscale context information is obtained to enhance feature representation, which helps boost the final performance. The decoder network recovers the fused features obtained from the cascaded PPM and realizes a precise segmentation result. The dice coefficient loss, rather than the CE loss or the MSE loss, is adopted to deal with the class imbalance issue of the road and the background. The quantitative and qualitative results on two datasets demonstrate that the proposed method can effectively handle occlusion areas and achieve homogenous and smooth road segmentation results.

## 3. Proposed method

In this section, we discuss the details of the proposed RDB, PPM, architecture of GC-DCNN, and loss function, which are the key elements of our proposed method.

### 3.1. RDB

In the traditional deep CNN model, a plain convolutional kernel, shown in Fig. 1(a), with a fixed size window slides over feature maps and transforms the spatial context information into high-level features with semantic information layer by layer. However, Zhou et al. [42] showed that the empirical receptive field of CNNs is smaller than the theoretical field, particularly on the deeper layers. Thus, many networks fail to sufficiently utilize a large range of spatial information (i.e., the global context), which may alleviate the occlusion issue and noise pixels in the road segmentation task. We design the RDB unit, shown in Fig. 1(c), on the basis of the residual block unit, shown in Fig. 1(b) [16], to enlarge the receptive field in the limited resolution reduction and utilize more spatial context. In contrast with the common unit used in many networks, we adopt full pre-activation [17] to build the RDB unit. As shown in Fig. 1(c), the RDB unit uses the feature maps (pre-activation) as input. After a batch normalization (BN) layer and a rectified linear unit (ReLU) activation layer, the first $3 \times 3$ convolutional layer without dilated operation is applied to generate the new feature maps, followed by two other com-

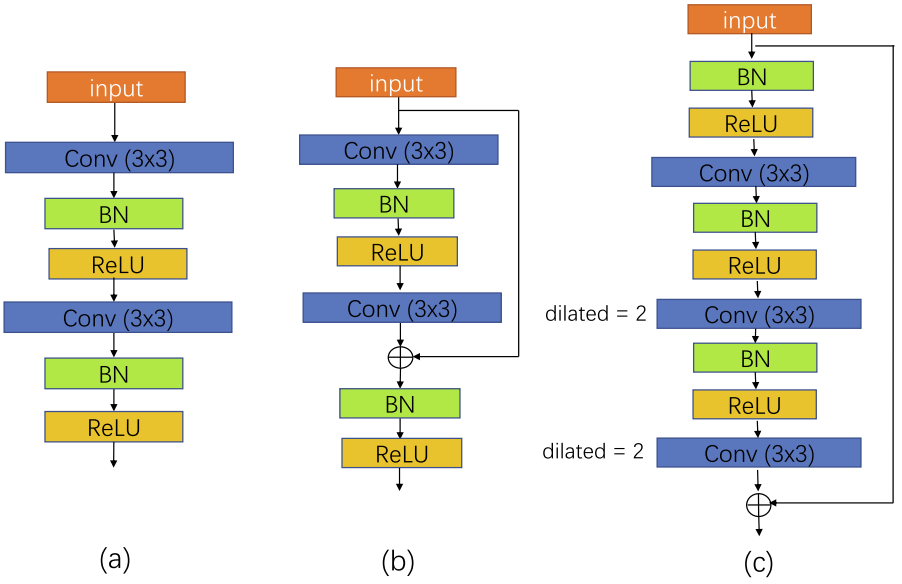


**Fig. 1.** Building blocks of CNN: (a) plain (b) residual, and (c) residual dilated blocks.

binations of BN, ReLU, and 3 × 3 convolutional layers. Therefore, the block has three 3 × 3 convolutional layers. To enlarge the effective receptive field of the block and utilize more spatial information, we use a dilated convolution operator. As shown in Fig. 2, this operator introduces sparsity into the convolution kernel and can apply the same filter at different ranges using various dilation ratios. We set the dilation ratio to 2 of the last two convolutional layers in the RDB to increase the receptive field of the final 3 × 3 convolutional layers from 7 to 11. Inspired by the residual block, we add the input and final feature maps before the unit output. This process is called shortcut connection. The designed structure facilitates the flow of information and the fusion of multiscale features. The RDB can be formulated as follows:

$$\mathbf{y}_l = h(\mathbf{x}_l) + \mathscr{F}(\mathbf{x}_l, \mathscr{W}_l)$$

where $\mathbf{y}_l$ and $\mathbf{x}_l$ are the output and input of the RDB unit respectively, $\mathscr{F}(\cdot)$ denotes the residual function and $h(\mathbf{x}_l)$ is a identity mapping function which typically is $h(\mathrm{x}_l) = \mathrm{x}_l$.

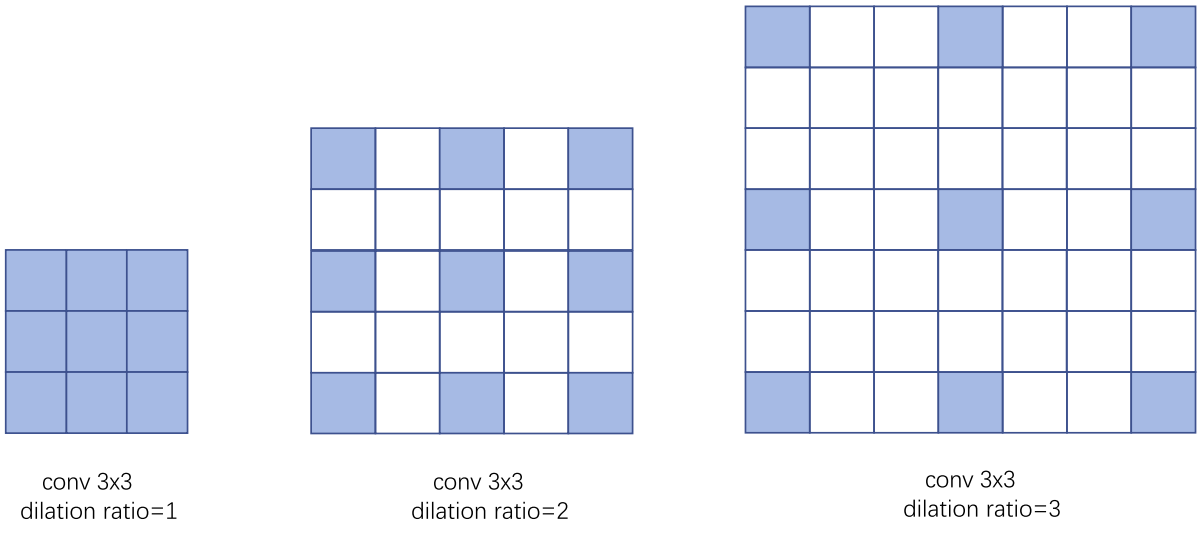


**Fig. 2.** Dilated convolution with different dilation ratios.

### 3.2. PPM

In deep CNNs, the global context information that can be used largely depends on the size of the receptive field of a network. Recent studies [22,41,5] have indicated that global context information exerts a significant impact on the performance of semantic segmentation. However, the effective receptive field of CNNs is always smaller than the theoretical case even if a deeper network is built [42]. Global average pooling is an efficient method for obtaining global context information, and it has been frequently used in image classification and semantic segmentation tasks [22]. Zhao et al. [41] proposed PPM, which used multilevel global average pooling to aggregate multigrained global features for scene parsing tasks. We use PPM on the final feature maps of the encoder network to obtain the global representation, allowing us to utilize multiscale global context information with limited network depth in road segmentation task.

As shown in Fig. 3, PPM has four level scales with bin sizes of 1, 2, 3, and 6. The first level is the global average pooling on the input feature map to generate a single bin. At the succeeding pyramid level, feature maps are divided into different subregions and pooled to produce the representation of different locations by using pooling kernels of different sizes. Suppose the height and width of the input features are H and W, respectively; the bin size of each level is N, and the sizes of the pooling kernels are H/N and W/N. The output of PPM preserves the global context information with different sizes due to the structure. A $1 \times 1$ convolutional layer is applied to each pyramid level to reduce the dimensions of global feature maps to 1/4 of the input and ensure that different-level features have the same weight. Then, the low-dimensional feature maps are upsampled to the same size as the original feature maps via bilinear interpolation. Lastly, the different-level and input features of PPM are concatenated as the final global context features. PPM can be formulated as follows:

$$\mathbf{y}_p = \mathrm{Concat}\left(\sigma_1(\mathbf{x}_p), \sigma_2(\mathbf{x}_p), \sigma_3(\mathbf{x}_p), \sigma_4(\mathbf{x}_p), \mathbf{x}_p\right)$$

where $\mathbf{y}_p$ and $\mathbf{x}_p$ are the output and input of PPM, respectively; $\sigma(\cdot)$ denotes the function, which successively consists of pooling, convolution transform, and upsampling; and $\mathrm{Concat}(\cdot)$ is the concatenation operation.

Notably, the number of pyramid levels and the bin size of each level can be modified. They are highly related to the size of feature maps that are fed into PPM. We can select these hyperparameters in accordance with the situation to maintain a reasonable gap in representation.

### 3.3. GC-DCNN architecture

Here, we illustrate in detail how the GC-DCNN with an RDB unit and PPM described earlier is built. As shown in Fig. 4, the GC-DCNN is a U-type network that is composed of an encoder network, PPM, and a decoder network. The encoder part extracts hierarchical features from the input image, and the global context information is aggregated by PPM. The decoder network restores the fused features to predict the pixel-level road and background areas.

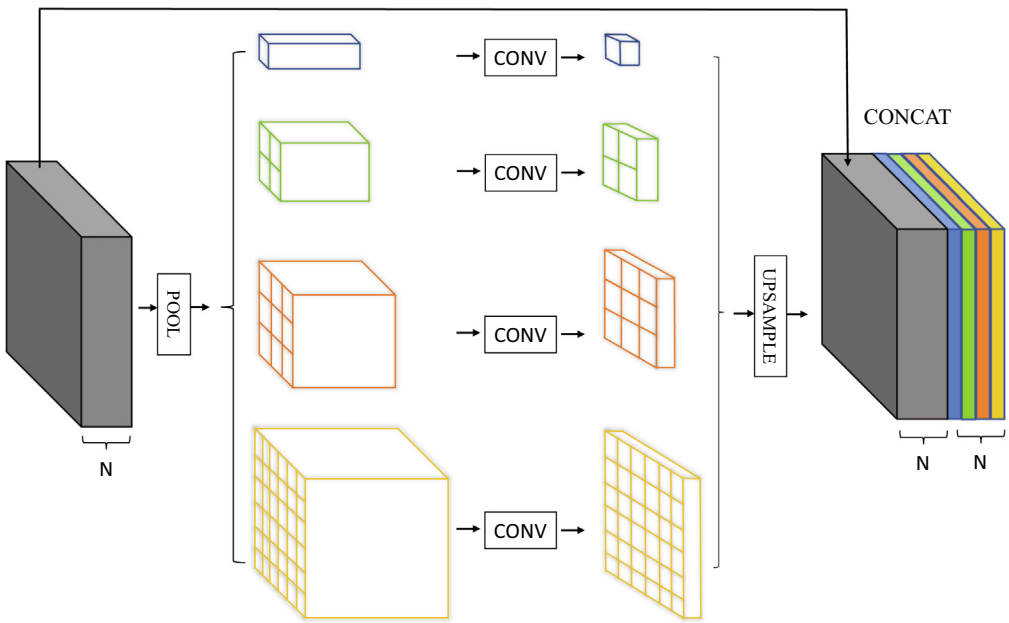


**Fig. 3.** Pyramid pooling module.

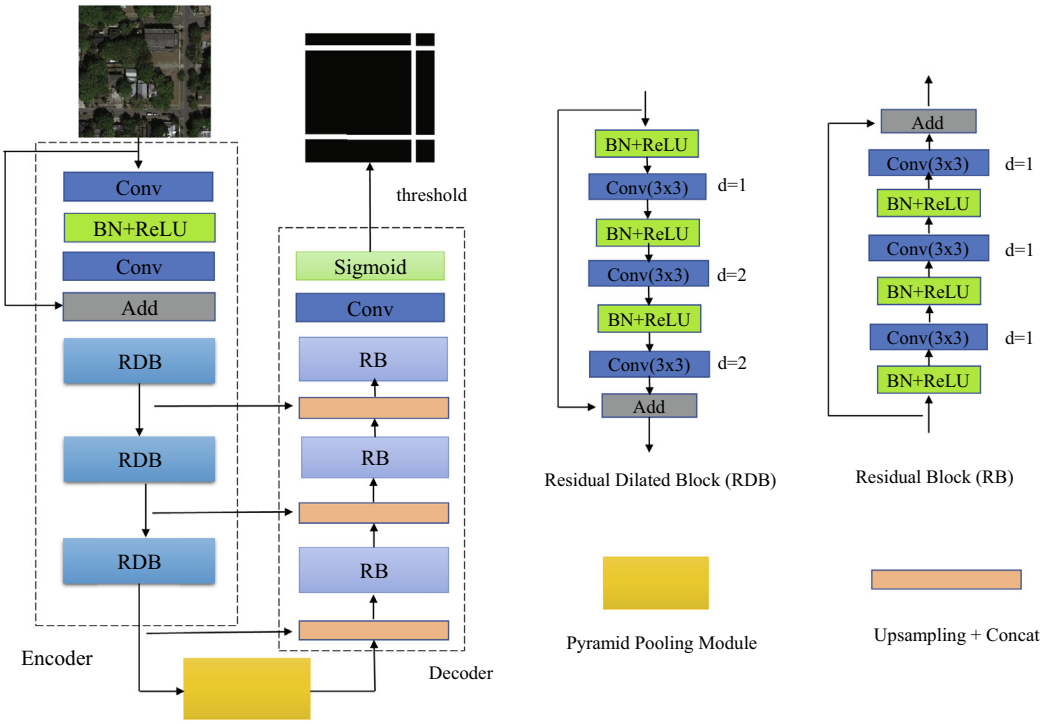


**Fig. 4.** Flowchart of the proposed deep GC-DCNN (left), which contains five parts: the input image, encoder, PPM, decoder and the output. The detailed architecture of the network is provided in Table 1.

Following the setting of CasNet [7], we design a smaller network for the road segmentation of remote sensing images rather than the commonly used segmentation network. GC-DCNN has 21 convolutional layers. In contrast with most previous studies [21,43] that only used dilated convolution in the bridge stage, we build the entire encoder network with RDBs. The encoder network consists of the two initial convolutional layers and three RDBs. The two initial convolutional layers convert the input RGB image into the primary high-dimensional features, and then the features are fed into subsequent blocks to generate the multiscale hierarchical feature. Instead of using the max pooling operation to downsample the feature maps, we set the stride of the first convolutional layer to 2 in each RDB unit. Therefore, the total stride of the encoder network is 8. PPM works similar to a bridge, which uses the final features of the encoder as the input and produces the features with global context representation for the decoder part. The decoder network is also composed of three special RDB units, whose dilation ratios are set to 1 for refinement. These block units are connected via the upsampling operation, which is implemented by the transposed convolution operator in Pytorch with a kernel size and stride of 2. The upsampled features in each level are concatenated with the corresponding hierarchical features of the encoder in the depth dimension to obtain features with rich spatial details. The last $1 \times 1$ convolutional layer converts the high-dimensional features into single-channel features, and the loss function is calculated with the ground truth after the sigmoid activation function. Table 1 provides the details of each layer, including the different parts of GC-DCNN, the number of layers, the hyperparameters (e.g., kernel size, output channels, stride, and dilated ratio of the convolutional layer), and the output size of each layer. Notably, we omit the BN layer and the ReLU activation function between the convolutional layers for brevity.

Compared with the conventional network for road segmentation, our proposed GC-DCNN exhibits the following advantages:

1) Under the same network architecture, the encoder network built by RDBs has a larger receptive field and captures more spatial information during feature extraction without additional parameters and computations.
2) The PPM embedded into the model generates and fuses multiscale global features based on input features, providing features with stronger representation ability for the decoder network.

### *3.4. Loss function*

Instead of adopting the loss function used in most existing methods [7,40,26] (e.g., CE loss), we adopt a more suitable loss function, called the dice coefficient loss, in our method. This function is inspired by medical image segmentation [24]. As shown in Fig. 5, a strong imbalance exists between the area of the road foreground and the complex background, similar

**Table 1**
Network structure of deep GC-DCNN.

| Name | Unit level | Layer | Filter | Stride | Dilated ratio | Output size |
|---|---|---|---|---|---|---|
| Input | | | | | | 256 × 256 × 3 |
| Encoder | Level 1 | Conv 1 | 3 × 3/64 | 1 | 1 | 256 × 256 × 64 |
| | | Conv 2 | 3 × 3/64 | 1 | 1 | 256 × 256 × 64 |
| | Level 2 | Conv 3 | 3 × 3/128 | 2 | 1 | 128 × 128 × 128 |
| | | Conv 4 | 3 × 3/128 | 1 | 2 | 128 × 128 × 128 |
| | | Conv 5 | 3 × 3/128 | 1 | 2 | 128 × 128 × 128 |
| | Level 3 | Conv 6 | 3 × 3/256 | 2 | 1 | 64 × 64 × 256 |
| | | Conv 7 | 3 × 3/256 | 1 | 2 | 64 × 64 × 256 |
| | | Conv 8 | 3 × 3/256 | 1 | 2 | 64 × 64 × 256 |
| | Level 4 | Conv 9 | 3 × 3/512 | 2 | 1 | 32 × 32 × 512 |
| | | Conv 10 | 3 × 3/512 | 1 | 2 | 32 × 32 × 512 |
| | | Conv 11 | 3 × 3/512 | 1 | 2 | 32 × 32 × 512 |
| PPM | Level 5 | | | | | 32 × 32 × 1024 |
| Decoder | Level 6 | Upsampling | 2 × 2/256 | 2 | 1 | 64 × 64 × 256 |
| | | Conv 12 | 3 × 3/256 | 1 | 1 | 64 × 64 × 256 |
| | | Conv 13 | 3 × 3/256 | 1 | 1 | 64 × 64 × 256 |
| | | Conv 14 | 3 × 3/256 | 1 | 1 | 64 × 64 × 256 |
| | Level 7 | Upsampling | 2 × 2/128 | 2 | | 128 × 128 × 128 |
| | | Conv 15 | 3 × 3/128 | 1 | 1 | 128 × 128 × 128 |
| | | Conv 16 | 3 × 3/128 | 1 | 1 | 128 × 128 × 128 |
| | | Conv 17 | 3 × 3/128 | 1 | 1 | 128 × 128 × 128 |
| | Level 8 | Upsampling | 2 × 2/64 | 2 | | 256 × 256 × 64 |
| | | Conv 18 | 3 × 3/64 | 1 | 1 | 256 × 256 × 64 |
| | | Conv 19 | 3 × 3/64 | 1 | 1 | 256 × 256 × 64 |
| | | Conv 20 | 3 × 3/64 | 1 | 1 | 256 × 256 × 64 |
| Output | | Conv 21 | 1 × 1/1 | 1 | 1 | 256 × 256 × 1 |

to several medical image segmentation tasks. If CE loss is used, then these small loss values can overwhelm the rare class (road pixels) when summing up numerous easy examples (background pixels). We utilize the dice coefficient loss function to guide the optimization of the proposed network and efficiently deal with the class imbalance issue. The extensive experimental results show that the performance of the dice coefficient loss is better.

The formulation of binary CE loss can be written as follows:

$$Loss = -\frac{1}{N}\sum_{i=1}^{N} y_i \log p_i + (1 - y_i) \log (1 - p_i)$$

where $N$ denotes the total number of pixels of the input image, $p_i$ indicates the predicted probability that this pixel is an expected pixel, and $y_i$ is the label of the ground truth of the ith pixel. $y_i = 1$ if the pixel belongs to the foreground, otherwise, $y_i = 0$.

The dice coefficient loss can be formulated as follows:

$$Loss = 1 - \frac{2TP}{2TP + FN + FP} = 1 - \frac{2\sum_{i}^{N} p_i g_i}{\sum_{i}^{N} p_i^2 + \sum_{i}^{N} g_i^2}$$

where TP denotes the true positive, FP denotes the false positive, and FN denotes the false negative. $p_i \in [0, 1]$ is the value of the $i$th pixel that belongs to the predicted binary segmentation mask, and $g_i$ is the value of the $i$th pixel that belongs to the ground truth binary mask. N is the total number of pixels in the predicted mask or ground truth mask. The loss function is optimized directly on the evaluation metric F1 score, as described in the experimental part.

## 4. Experiments

In this section, we evaluate the proposed GC-DCNN algorithm on two different publicly available road datasets to demonstrate its superiority. Seven comparative methods, namely, FCN [23], UNet [29], ResUNet [40], CasNet [7], RoadCNN [3], RCNN-UNet [35] and Topo-UNet [25] are selected as baselines. The visual and quantitative segmentation results are presented. We perform ablation experiments to prove the effectiveness of the proposed RDB and introduced PPM. We also investigate the influences of different loss functions on road segmentation performance.

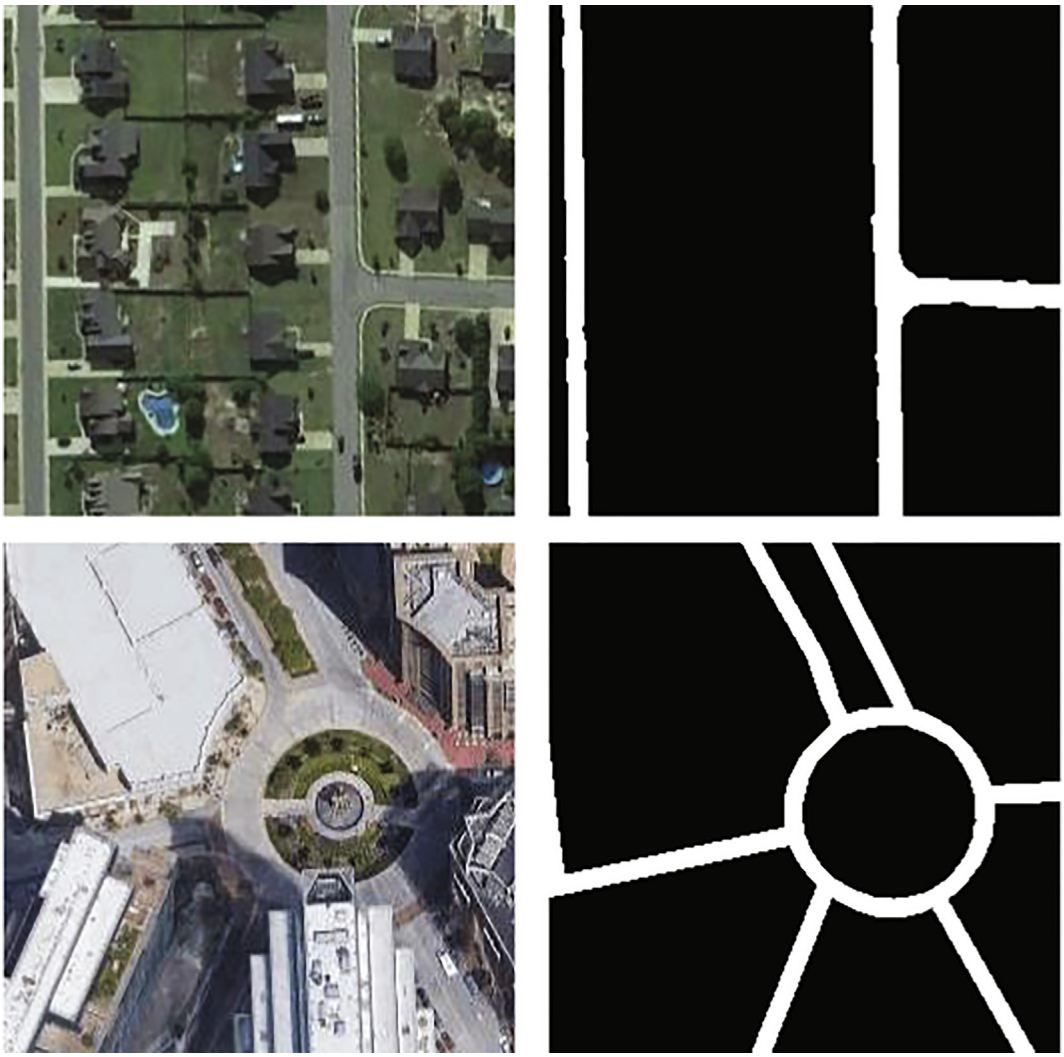

**Fig. 5.** The original images and corresponding labels in two different dataset. The areas of road and background are imbalanced.

### 4.1. Experimental settings

#### 4.1.1. Datasets

.

1) *CasNet dataset (CNDS)*: CNDS, built by Cheng et al. [7], contains 224 VHR images collected from Google Earth. The size of each image in the dataset is at least 600 × 600 pixels, and the spatial resolution is 1.2 m per pixel. Most images contain complex backgrounds and occlusions caused by trees or cars, making the road segmentation task challenging. We follow the setting of [7] and randomly select 180 images as the training set and 14 images as the validation set. The remaining 30 images are used as the test set.
2) *Roadtracer dataset (RTDS)*: This dataset was created and first used in [3]. RTDS is a large corpus of high-resolution satellite images and ground truth road network graphs covering the urban core of 40 cities in 6 countries. In each city, approximately 24 $km^2$ of the center area is selected as the sample of the dataset for a total of 300 images with a resolution of 4096 × 4096. Following [3], images from 25 cities are randomly selected for training, while the test set contains images from the 15 remaining cities.

#### 4.1.2. Data preprocessing and augmentation

We cannot directly train the models with the original images due to the high resolution of remote sensing images and the limited GPU resource. Moreover, the number of samples in the original dataset is insufficient for training these deep

learning-based models. Therefore, following previous studies [7,3], data preprocessing and augmentation are adopted before model training.

1) *CNDS*: Following [7], given an image in the training and validation sets, we first crop 5 fixed-position patches (4 corner patches and the center one) and another 15 patches selected randomly from the free position of the image. The size of the patches is 256 × 256. We flip each patch in the horizontal direction and rotate the original and flipped patches at a step of 90° 4 times. Thus, the size of the training and validation sets is increased by a factor of 160 (20 × 2 × 4). For the test set, we perform cropping operation without rotation.
2) *RTDS*: We use the same setting as [3] and divide all the 4096 × 4096 images in the dataset into 256 patches with size 256 × 256.

We filter the samples with less than 1000 pixels of road area in both datasets to effectively calculate loss in training and accuracy in testing. Finally, 28,420 and 590 samples remain in the training and test sets of CNDS, respectively. The number of images in the training and test se of RTDS is 102,212 and 12,936, respectively.

*4.1.3. Evaluation metrics*

Four common metrics, namely, completeness (COM), correctness (COR), quality (Q), and F1 score, are used to evaluate the quantitative performance of road segmentation. COM measures the percentage of matched areas in the ground truth map. COR represents the proportion of matched road areas in the predicted segmentation map. Q combines COM and COR. The F1 score is a harmonic average between COM and COR that can measure the robustness of methods. The four metrics are formulated as follows:

$$\mathrm{COM} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \qquad \mathrm{COR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}},$$

$$\mathrm{Q} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP}}, \quad \mathrm{F1} = \frac{2 \times \mathrm{COM} \times \mathrm{COR}}{\mathrm{COM} + \mathrm{COR}}$$

where TP, FP and FN denote the true positive, false positive and false negative respectively. Notably, a larger metric value indicates better performance.

In addition, two efficiency metrics, namely, Params (the number of model parameters) and speed (the running speed of the model), are included to measure processing efficiency.

*4.1.4. Implementation details*

The proposed GC-DCNN is implemented using the Pytorch framework and trained on 4 NVIDIA TitanXp GPU in a distributed manner. During training, images with size 256 × 256 are randomly sampled from the dataset and then fed into the network. Given the limited GPU memory, the minibatch size is set as 12 for each GPU, and the Adam optimizer with betas of 0.9 and 0.999 is adopted to optimize the GC-DCNN model with the original learning rate of 0.001. We train the model for 50 epochs and drop the learning rate by a factor of 0.1 at 10 and 40 epochs. In the inference process, we set the threshold at 0.5, indicating that the final value of each position in the output probability map is 1 if the predicted value is greater than 0.5 and 0 otherwise.

*4.1.5. Comparison methods*

Comparison methods can be divided into two major categories: two baselines for the general image segmentation task and five improved methods for the road segmentation task. In addition, considering the fairness for all the methods, we slightly adjust the backbone networks of several baselines for the same feature extraction ability. We adopt the original architectures of several baselines, including CasNet, ResUNet, RoadCNN, and RCNN-UNet and make no changes. Here, we briefly describe the comparison methods as follows.

1) *FCN*: FCN is the first work to train a fully convolutional network end-to-end for pixelwise prediction and achieve state-of-the-art performance during that time. FCN takes an arbitrary sized image as input and has been used as a solid baseline in many semantic segmentation studies.
2) *UNet*: UNet was initially proposed for the biomedical image segmentation, which has an encoderdecoder structure. UNet has been extended to many other segmentation tasks due to its sophisticated structure and excellent performance. We construct a similar encoder network for UNet in this experiment to keep the same feature extraction ability as other methods.
3) *CasNet*: Cheng et al. [7] proposed the cascaded end-to-end CNN for road segmentation and centerline extraction. We select the network for the road segmentation part, which has the symmetrical encoderdecoder structure without skip connection for 20 convolutional layers in CasNet.
4) *ResUNet*: ResUNet combines the advantages of UNet and residual learning, promoting information propagation among different-level features and making the training easy.

5) *RoadCNN*: RoadCNN is the deep learning-based baseline in [3]. RoadCNN consists of 20 convolutional layers and over 512 channel layers with dropout.
6) *RCNN-UNet*: RCNN-UNet designs an RCNN unit, in which convolutional filters share weights to build a deeper network to better utilize the spatial context.
7) *Topo-UNet*: In this method, a new loss function, called topology-aware loss, is proposed as a penalty term with binary CE loss to account for the topology information. Here, we adopt the UNet mentioned in 2) as the encoderdecoder network for a fair comparison.

### 4.2. Performance evaluation

#### 4.2.1. Evaluation on CNDS

We firstly evaluate the performance of all the methods on the CNDS test set. The quantitative comparisons are reported in Table 2, and the visual segmentation results of different methods are presented in Fig. 6.

In Table 2, the best performance of each evaluation metrics is highlighted in bold. GC-DCNN outperforms all the compared methods in all the four metrics. The overall accuracies of all the methods are relatively high and comparable because the roads in CNDS are relatively neat and the background is not too complex. The basic FCN model has the lowest accuracy because the predicted segmentation result is directly upsampled from high-level feature maps without refinement. The other methods based on the encoderdecoder structure perform better because the refinement process and feature fusion strategy provide richer spatial detail information. Although RoadCNN exhibits a performance similar to that of GC-DCNN in COM, its COR is unsatisfactory. Compared with basic UNet, Topo-UNet, which is optimized with topology-aware loss, achieves performance improvement. Our proposed GC-DCNN obtains the highest F1 score, demonstrating its stronger robustness and generalization compared with the other methods. With regard to processing efficiency, the simple methods have less parameters and faster speed. Meanwhile, our GC-DCNN achieves the best performance with competitive parameters and speed.

Fig. 6 presents the visual road segmentation results of these methods on the same test image. All the compared methods do not perform well in dealing with the occlusion problem caused by the tree (as highlighted in the sub-images) marked in the segmentation results. By contrast, GC-DCNN solves this problem efficiently. This result can be attributed to the rich spatial information provided by the larger receptive field of RDBs during the encoder process and the global context information generated by PPM.

#### 4.2.2. Evaluation on RTDS

The quantitative performance and visual performance of all the methods on RTDS are provided in Table 3 and Fig. 7, respectively.

Table 3 shows that GC-DCNN significantly outperforms all the compared methods in COM, but exhibits ordinary performance in COR, indicating that GC-DCNN finds more true positive pixels with only a few additional false positive pixels. The highest values in Q and F1 also demonstrate that GC-DCNN achieves a more balanced and robust performance compared with the other segmentation algorithms. Given the complex urban background, all the methods provide relatively low accuracy compared with the CNDS results. The efficiency of all the methods is the same as that for CNDS because we simply change the evaluation dataset.

As shown in Fig. 7, the road segmentation results of FCN, UNet, CasNet, ResUNet, and RCNN-UNet are irregular due to the occlusion of surrounding trees and several mispredicted road regions among all the baseline methods. The segmentation result of GC-DCNN is smoother and more similar to the ground truth.

### 4.3. Ablation study

#### 4.3.1. Ablation study of RDBs

Table 4 shows the results of GC-DCNN with different dilated convolution configurations. The encoder constructed with stacked RDBs outperforms the encoder that typically uses dilated convolution in the bridge stage of UNet [21,43]. The result proves that the encoder of GC-DCNN can extract more representative features than the latter in a road segmentation task.

**Table 2**
Evaluation results of different methods on the CNDS test set. The measure of time is seconds per image.

| Method | COM | COR | Q | F1 | Params | Speed |
|---|---|---|---|---|---|---|
| FCN | 91.36 | 93.75 | 86.12 | 92.34 | 29.14 M | 0.003 s |
| UNet | 91.87 | 93.93 | 86.75 | 92.75 | 35.66 M | 0.009 s |
| CasNet | 91.85 | 94.27 | 86.96 | 92.90 | 26.30 M | 0.009 s |
| ResUNet | 91.87 | 94.45 | 87.16 | 92.99 | 30.75 M | 0.011 s |
| RoadCNN | 92.85 | 94.57 | 88.15 | 93.57 | 77.15 M | 0.025 s |
| RCNN-UNet | 92.30 | 94.86 | 87.86 | 93.44 | 30.48 M | 0.018 s |
| Topo-UNet | 92.56 | 95.15 | 88.36 | 93.71 | 35.66 M | 0.010 s |
| GC-DCNN | **92.88** | **95.40** | **88.87** | **94.02** | 38.57 M | 0.013 s |

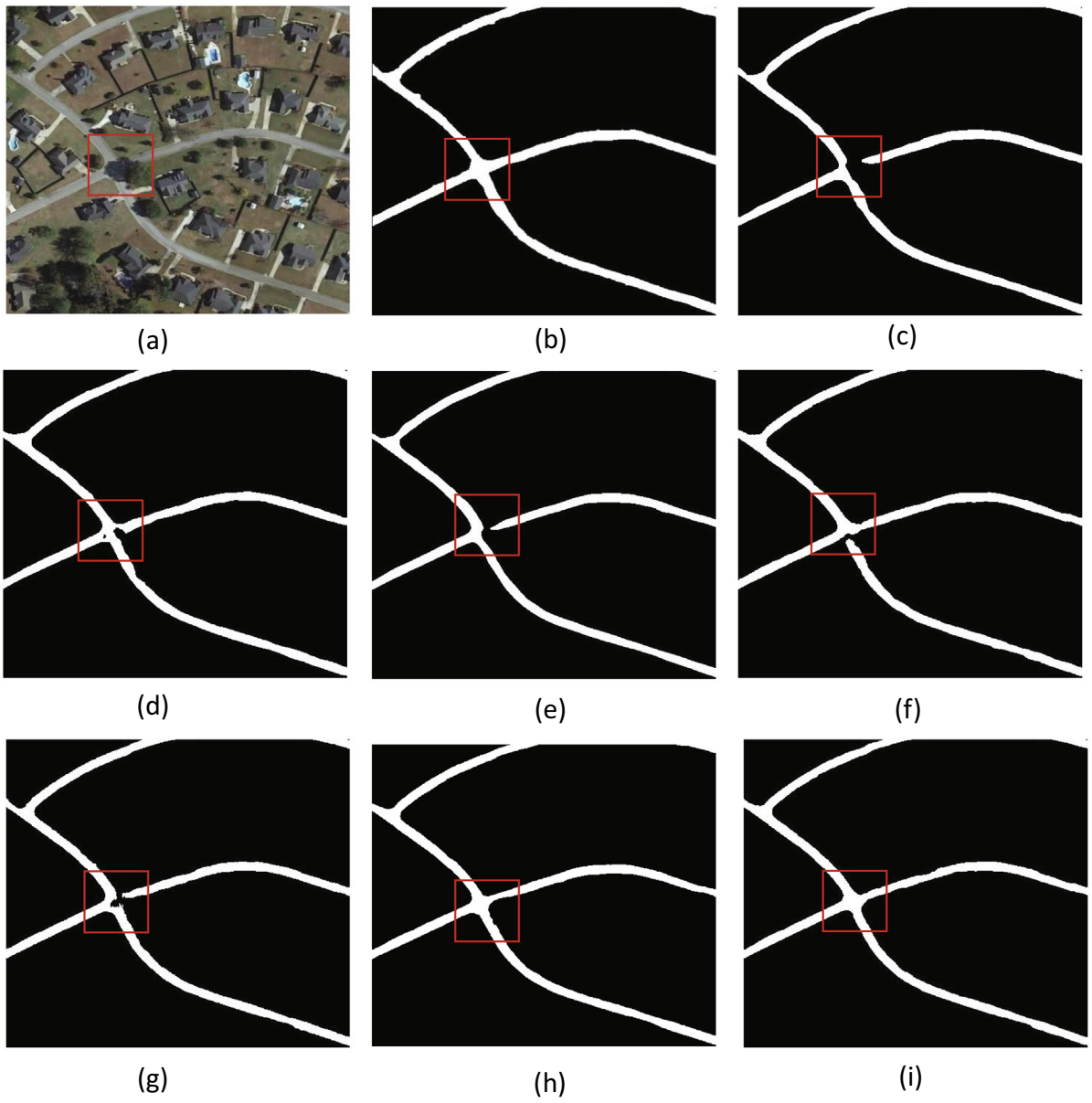


**Fig. 6.** Visualization of road segmentation results on a test sample from CNDS. a: Original test image. b: Corresponding ground truth of image. c: Segmentation result of FCN. d: Segmentation result of UNet. e: Segmentation result of CasNet. f: Segmentation result of ResUNet. g: Segmentation result of RoadCNN. h: Segmentation result of RCNN-UNet. i: Segmentation result of our proposed GC-DCNN.

**Table 3**
Evaluation results of different methods on the RTDS test set.

| Method | COM | COR | Q | F1 | Params | Speed |
|---|---|---|---|---|---|---|
| FCN | 50.37 | 60.17 | 39.81 | 54.84 | 29.14 M | 0.003 s |
| UNet | 50.79 | 64.85 | 40.45 | 55.11 | 35.66 M | 0.009 s |
| CasNet | 57.85 | 62.16 | 43.93 | 58.61 | 26.30 M | 0.009 s |
| ResUNet | 53.55 | 60.35 | 40.38 | 55.38 | 30.75 M | 0.011 s |
| RoadCNN | 61.39 | **66.34** | 48.19 | 62.47 | 77.15 M | 0.025 s |
| RCNN-UNet | 58.58 | 62.69 | 44.60 | 59.20 | 30.48 M | 0.018 s |
| Topo-UNet | 59.27 | 65.11 | 46.50 | 61.02 | 35.66 M | 0.010 s |
| GC-DCNN | **67.18** | 64.62 | **50.24** | **64.59** | 38.57 M | 0.013 s |

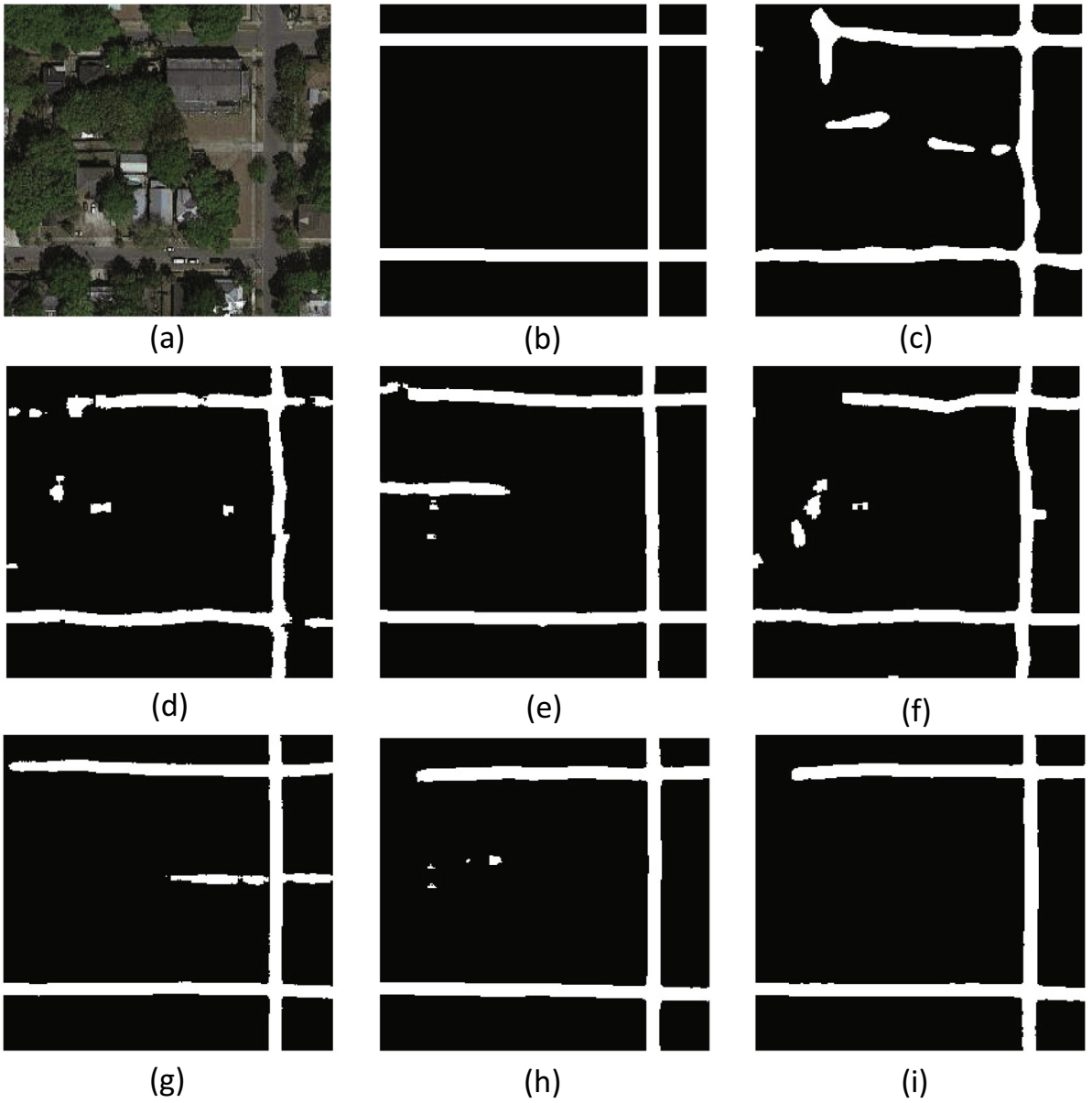


**Fig. 7.** Visual results of all the road segmentation methods on a test image from RTDS. a: Original test image. b: Corresponding ground truth of image. c: Segmentation result of FCN. d: Segmentation result of UNet. e: Segmentation result of CasNet. f: Segmentation result of ResUNet. g: Segmentation result of RoadCNN. h: Segmentation result of RCNN-UNet. i: Segmentation result of our proposed GC-DCNN.

**Table 4**
Investigation of the influence of the placement of dilated convolution. Road segmentation results on the test set of RTDS.

| Method | | COM | COR | Q | F1 |
|---|---|---|---|---|---|
| GC-DCNN | Bridge | 65.97 | 64.64 | 49.65 | 64.08 |
| | RDB | **67.18** | **64.62** | **50.24** | **64.59** |

When designing the RDB unit, we notice that different dilation ratio combinations exert varying effects on the final results. To obtain the best performance, we conduct an ablation experiment to determine the optimal combination of dilation ratios. Given that the first convolutional layer with a stride of 2 downsamples the resolution, we fix its dilation ratio at 1 and only discuss the last 2 convolutional layers of the RDB unit. We arrange the combinations of dilation ratios in ascending order. The road segmentation results of GC-DCNN with different dilation ratios on RTDS are provided in Table 5. The best

**Table 5**
Results of GC-DCNN with different dilation ratios on the test set of RTDS. Here, we only present the dilation ratio of last two convolutional layers.

| method | Dilated Ratio | COM | COR | Q | F1 |
|---|---|---|---|---|---|
| GC-DCNN | (1,1) | **67.48** | 63.57 | 49.68 | 64.20 |
| | (1,2) | 66.85 | 64.54 | 50.02 | 64.45 |
| | (1,3) | 66.42 | **64.95** | 50.10 | 64.44 |
| | (**2**, **2**) | 67.18 | 64.62 | **50.24** | **64.59** |
| | (2,3) | 66.84 | 64.45 | 49.97 | 64.34 |
| | (3,3) | 66.88 | 64.27 | 49.84 | 64.20 |

performance of COM and COR is the combination of (1,1) and (1,3), respectively. Meanwhile, GC-DCNN with a dilation ratio combination of (2,2) achieves the best comprehensive Q and F1 scores. Different dilation ratio combinations produce various feature representations, and the results show that (2,2) is the optimal combination for this task.

*4.3.2. Ablation study of PPM*

We perform an experiment on RTDS to investigate the effect of PPM, which produces the core global context feature in GC-DCNN. The results are presented in Table 6. The network without PPM works poorly in all the metrics, whereas the network with PPM exhibits more comprehensive advantages, indicating that the global context information plays an important role in improving road segmentation performance. Fig. 8 presents the visual comparison results of GC-DCNN with PPM. The model without PPM incorrectly predicts some background areas as roads, and the predicted road region is not as smooth as that predicted by GC-DCNN with PPM.

*4.3.3. Ablation study for loss function*

Lastly, we discuss the impact of using different loss functions on the performance of GC-DCNN on CNDS and RTDS. Table 7 indicates that the model optimized using the dice coefficient loss exhibits the best performance on both datasets. For the simpler dataset CNDS, the segmentation results of GC-DCNN with the two different loss functions are similar, but the model based on the dice coefficient loss performs better on all the metrics. For the more challenging dataset, RTDS, in which road pixels have more complex background, GC-DCNN with the dice coefficient loss shows significant improvement in the COM metrics and approximately 3.6% higher in F1 score compared with the model with BCE loss but slightly worse performance in

**Table 6**
Investigation of the influence of PPM. Road segmentation results on the RTDS test set.

| Method | PPM | COM | COR | Q | F1 |
|---|---|---|---|---|---|
| GC-DCNN | no | 65.32 | 64.32 | 49.08 | 63.47 |
| | yes | **67.18** | **64.62** | **50.24** | **64.59** |


**Fig. 8.** Visual results on a test image from RTDS. a: GC-DCNN without PPM. b: GC-DCNN with PPM.

**Table 7**
Road segmentation results with different loss functions on the CNDS and RTDS test sets. BCE: binary cross entropy loss. Dice: dice coefficient loss.

| Method | Dataset | Loss function | COM | COR | Q | F1 |
|---|---|---|---|---|---|---|
| GC-DCNN | CNDS | BCE | 92.25 | 94.86 | 87.86 | 93.42 |
| | | Dice | **92.88** | **95.40** | **88.87** | **94.02** |
| | RTDS | BCE | 59.72 | **65.11** | 46.50 | 61.02 |
| | | Dice | **67.18** | 64.62 | **50.24** | **64.59** |

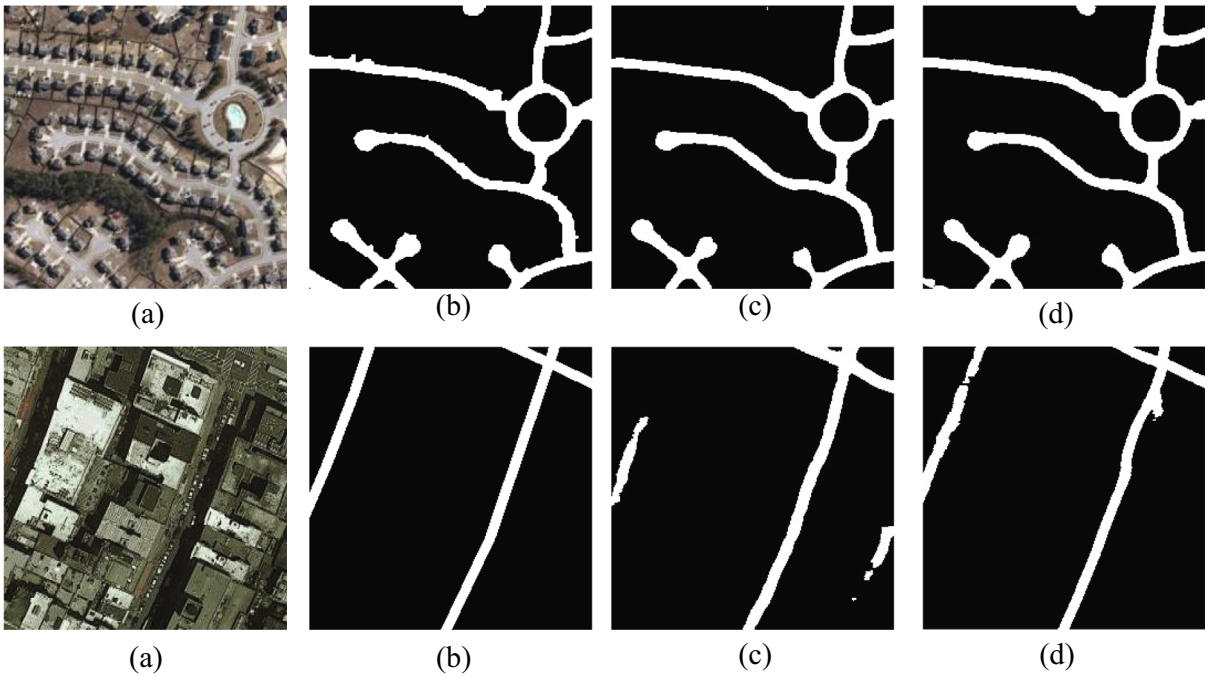


**Fig. 9.** Visual results on a test image from CNDS and RTDS. The first row of image belongs to CNDS and the second row belongs to RTDS. a: The original image in the test set. b: Corresponding ground truth. c: Visual result of GC-DCNN with BCE loss. d: Visual result of GC-DCNN with dice coefficient loss.

COR. The preceding experimental results on the two different datasets demonstrate that GC-DCNN that is directly optimized on the evaluation metrics is better in solving the occlusion problem and distinguishing between road regions and background areas. The visualization results of GC-DCNN with different loss functions on both datasets are presented in Fig. 9, which shows similar experiment results.

## 5. Conclusion

In this study, we propose a novel GC-DCNN for the road segmentation of remote sensing images. GC-DCNN exhibits the following advantages. 1) We design RDBs to enlarge the receptive field, such that the encoder network built with RDBs can produce more discriminative features with high resolution. The residual connection strategy in a block facilitates the flow of information and eases deep network training. 2) PPM is used to capture multiscale global context information and generate features with rich representation. The designed RDB and PPM aim to enhance feature representation, which is crucial for solving the occlusion problem and improving segmentation performance. 3) The dice coefficient loss is adopted as the loss function to optimize the network. This loss can efficiently address the class imbalance problem in a binary segmentation task. Extensive experiments are conducted on two real-world benchmark road segmentation datasets. The results show that our proposed GC-DCNN method achieves state-of-the-art performance.

## CRediT authorship contribution statement

**Meng Lan**: Writing - original draft, Methodology, Conceptualization. **Yipeng Zhang**: Investigation. **Lefei Zhang**: Writing - review & editing. **Bo Du**: Validation.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under grants 61822113 and 61771349, in part by the Natural Science Foundation of Hubei Province under grant 2018CFB432, and in part by the Science and Technology Major Project of Hubei Province (Next-Generation AI Technologies) under Grant 2019AEA170. The numerical calculations in this paper had been supported by the supercomputing system in the Supercomputing Center of Wuhan University.

## References

[1] J.M. Alvarez, T. Gevers, Y. LeCun, A.M. Lopez, Road scene segmentation from a single image, in: Proceedings of European Conference on Computer Vision, 2012, pp. 376–389..
[2] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: a deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2017) 2481–2495.
[3] F. Bastani, S. He, S. Abbar, M. Alizadeh, H. Balakrishnan, Roadtracer: Automatic extraction of road networks from aerial images, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2018, pp. 4720–4728..
[4] F. Cao, Y. Liu, D. Wang, Efficient saliency detection using convolutional neural networks with feature selection, Inf. Sci. 456 (2018) 34–49.
[5] L. Chen, G. Papandreou, F. Schroff, H. Adam, Rethinking atrous convolution for semantic image segmentation abs/1706.05587, 2017a. URL:http://arxiv.org/abs/1706.05587.
[6] L.C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (2017) 834–848.
[7] G. Cheng, Y. Wang, S. Xu, H. Wang, S. Xiang, C. Pan, Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network, IEEE Trans. Geosci. Remote Sens. 55 (2017) 3322–3337.
[8] G. Cheng, P. Zhou, J. Han, Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images, IEEE Trans. Geosci. Remote Sens. 54 (2016) 7405–7415.
[9] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078, 2014..
[10] G. Dartmann, H. Song, A. Schmeink, Big Data Analytics for Cyber-physical Systems: Machine Learning for the Internet of Things, Elsevier, 2019.
[11] P. Esser, E. Sutter, B. Ommer, A variational u-net for conditional appearance and shape generation, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2018, pp. 8857–8866..
[12] C. Farabet, C. Couprie, L. Najman, Y. LeCun, Learning hierarchical features for scene labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 1915–1929.
[13] J. Han, D. Zhang, G. Cheng, L. Guo, J. Ren, Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning, IEEE Trans. Geosci. Remote Sens. 53 (2014) 3325–3337.
[14] B. Hariharan, P. Arbeaez, R. Girshick, J. Malik, Simultaneous detection and segmentation, in: Proceedings of European Conference on Computer Vision, 2014, pp. 297–312..
[15] K. He, J. Sun, X. Tang, Single image haze removal using dark channel prior, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2010) 2341–2353.
[16] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2016a, pp. 770–778..
[17] K. He, X. Zhang, S. Ren, J. Sun, Identity mappings in deep residual networks, in: Proceedings of European Conference on Computer Vision, 2016b, pp. 630–645..
[18] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of Advances in Neural Information Processing Systems, 2012, pp. 1097–1105..
[19] Q. Li, L. Chen, M. Li, S.L. Shaw, A. Nuchter, A sensorfusion drivable-region and lane-detection system for autonomous vehicle navigation in challenging road scenarios, IEEE Trans. Vehicular Technol. 63 (2014) 540–555.
[20] X. Li, H. Chen, X. Qi, Q. Dou, C.W. Fu, P.A. Heng, H-denseunet: hybrid densely connected unet for liver and tumor segmentation from ct volumes, IEEE Trans. Med. Imag. 37 (2018) 2663–2674.
[21] Z. Li, C. Peng, G. Yu, X. Zhang, Y. Deng, J. Sun, Detnet: A backbone network for object detection, 2018b. URL:http://arxiv.org/abs/1804.06215..
[22] W. Liu, A. Rabinovich, A.C. Berg, Parsenet: Looking wider to see better abs/1506.04579, 2015. URL:http://arxiv.org/abs/1506.04579.
[23] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431–3440..
[24] F. Milletari, N. Navab, S.A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: Proceedings of International Conference on 3D Vision, 2016, pp. 565–571..
[25] A. Mosinska, P. Mrquez-Neila, M. Kozinski, P. Fua, Beyond the pixel-wise loss for topology-aware delineation, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2018, pp. 3136–3145.
[26] G. Mttyus, W. Luo, R. Urtasun, Deeproadmapper: Extracting road topology from aerial images, Proceedings of International Conference on Computer Vision (2017) 3458–3466.
[27] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation, in: Proceedings of International Conference on Computer Vision, 2015, pp. 1520–1528..
[28] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards realtime object detection with region proposal networks, in: Proceedings of Advances in Neural Information Processing Systems, 2015, pp. 91–99.
[29] O. Ronneberger, P. Fischer, T. Brox, U-net: convolutional networks for biomedical image segmentation, in: Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention, 2015, pp. 234–241..
[30] S. Saito, T. Yamashita, Y. Aoki, Multiple object extraction from aerial imagery with convolutional neural networks, Electron. Imag. 2016 (2016) 1–9.
[31] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of International Conference on Learning Representations, 2015..
[32] L. Song, C. Wang, L. Zhang, B. Du, Q. Zhang, C. Huang, X. Wang, Unsupervised domain adaptive re-identification: theory and practice, Pattern Recognit. 102 (2020) 1–11, 107173.
[33] J.H. Tan, H. Fujita, S. Sivaprasad, S.V. Bhandary, A.K. Rao, K.C. Chua, U.R. Acharya, Automated segmentation of exudates, haemorrhages, microaneurysms using single convolutional neural network, Inf. Sci. 420 (2017) 66–76.

[34] C. Unsalan, B. Sirmacek, Road network detection using probabilistic and graph theoretical methods, IEEE Trans. Geosci. Remote Sens. 50 (2012) 4441–4453.
[35] X. Yang, X. Li, Y. Ye, R.Y.K. Lau, X. Zhang, X. Huang, Road detection and centerline extraction via deep recurrent convolutional neural network u-net, IEEE Trans. Geosci. Remote Sens. 57 (2019) 7209–7220.
[36] F. Yu, V. Koltun, T.A. Funkhouser, Dilated residual networks, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21–26, 2017, pp. 636–644..
[37] J. Yuan, D. Wang, B. Wu, L. Yan, R. Li, Legion-based automatic road extraction from satellite imagery, IEEE Trans. Geosci. Remote Sens. 49 (2011) 4528–4538.
[38] L. Zhang, L. Zhang, B. Du, Deep learning for remote sensing data: a technical tutorial on the state of the art, IEEE Geosci. Remote Sens. Mag. 4 (2016) 22–40.
[39] L. Zhang, L. Zhang, B. Du, J. You, D. Tao, Hyperspectral image unsupervised classification by robust manifold matrix factorization, Inf. Sci.s 485 (2019) 154–169.
[40] Z. Zhang, Q. Liu, Y. Wang, Road extraction by deep residual u-net, IEEE Geosci. Remote Sens. Lett. 15 (2018) 749–753.
[41] H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, Pyramid scene parsing network, in: Proceedings of Conference on Computer Vision and Pattern Recognition, 2017, pp. 6230–6239..
[42] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Semantic understanding of scenes through the ade20k dataset, Int. J. Comput. Vis. 127 (2019) 302–321.
[43] L. Zhou, C. Zhang, M. Wu, D-linknet: Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction, in: Proceedings of Conference on Computer Vision and Pattern Recognition Workshop, 2018, pp. 182–186..