Semantic Segmentation of Remote Sensing Images via Stepwise-Refined Large-Kernel Deconvolutional Networks

To cite this article: Zhao Xinwei et al 2019 J. Phys.: Conf. Ser. 1267 012050

View the article online for updates and enhancements.

This content was downloaded from IP address 207.241.226.230 on 14/11/2019 at 08:56
Semantic Segmentation of Remote Sensing Images via Stepwise-Refined Large-Kernel Deconvolutional Networks

Zhao Xinwei¹, Li Haichang², Wang Rui², Zheng Changwen² and Shi Song³

¹ Institute of Software Chinese Academy of Sciences, University of Chinese Academy of Sciences, Beijing, China
² Science and Technology on Integrated Information System Laboratory, Institute of Software Chinese Academy of Sciences, Beijing, China
³ Teleware Info & Tech (Fujian) Co.,LTD, Fujian Province, China

E-mail: rayarrow@qq.com

Abstract. Deep CNN based semantic segmentation has been developed for several years and many models are proposed. However, most of them are designed for natural scene images such as PASCAL VOC, and cannot perform very well on remote sensing images, in which objects are much smaller and more densely distributed than those in natural scene images. In this paper, we demonstrate the importance of high-resolution feature maps and the problem of large dilated convolutional kernels in semantic segmentation of remote sensing images. Furthermore, we propose a Stepwise-Refined Large-Kernel Deconvolutional Network with a focus on small and densely-distributed objects such as houses and buildings, or long and narrow ones such as roads and rivers. Experiments on a public available ISPRS Vaihingen Challenge Dataset and our self-compiled Fujian Dataset show that our model outperforms the state-of-the-art models in semantic segmentation of remote sensing images.

1. Introduction
Semantic segmentation is an important topic in computer vision, which can be seen as a per-pixel classification task. Specifically, given an image, we need to classify each pixel within a fixed set of classes, and produce a segmentation mask at last. See figure 1. Semantic segmentation has been widely applied in many fields such as autonomous vehicles, medical image analysis and land cover monitoring.

For image classification task, only semantic information is needed. But in semantic segmentation, spatial information is also required, in ordered to locate objects and recover their boundaries. Although deep learning based methods for semantic segmentation have been developed for many years, localization still remains a tough task especially for boundaries of objects.

Most proposed networks for semantic segmentation employs an encoder-decoder structure, in which encoders (such VGG16 [1] and ResNet [2]) are used to extract spatial and semantic information, while decoders are used to gather them to recover the results. FCN [3] (Fully Convolutional Neural network) was the first model trained end-to-end to perform the dense pixel-wise prediction tasks. Although FCN largely speeded up the training process and improved the accuracy, it cannot avoid the loss of spatial information caused by pooling layers, and this is the major reason to restrict the performance.

To tackle this problem, [4] proposed dilated convolution to replace pooling and standard non-dilated convolutional layers, which in turn expands the receptive fields in addition to preserving the resolution.
of feature maps without increasing the number of parameters. U-Net [5] suggests concatenating low-level features to high-level ones to compensate for the loss of spatial information. Deeplab v3 [6] and PSPNet [7] use pyramid pooling to capture multi-scale contextual information. Based on Deeplab v3, Deeplab v3+ [8] merged low-level feature maps to the pyramid pooling module like U-Net.

Among these models, Deeplab v3+ achieves state-of-the-art performance on PASCAL VOC dataset. It captures multi-scale contextual information at the highest layer of the encoder using ASPP (Atrous Spatial Pyramid Pooling), which employs 6-rated, 12-rated and 18-rated dilated kernels. However, objects in remote sensing images are small and densely-distributed, so regions covered by the kernels of ASPP often share similar patterns and not much discriminative semantic information can be captured. Worse still, noise are introduced occasionally. An example will be given to further illustrate this problem.

![Figure 1](image1.jpg)

**Figure 1.** Effects of 6-dilated convolution on 16-stride feature maps ignoring the effect of receptive field. Axes are kept on purpose to measure the sizes of objects. (b) and (d) are the corresponding segmentation mask of (a) and (c). (a) and (b) come from PASCAL VOC while (c) and (d) come from Fujian Dataset. Red rectangles are approximate visualization of the 6-rate dilated convolutional kernel of ASPP on 16-stride feature maps and yellow stars are non-zero weights in the kernel. The 6-dilated convolutional kernels are up sampled 16 times instead of down sampling original images by 16 times, as 16-time down sampled images are too blur and objects are hard to be identified.

In figure 1(a), it’s not easy to tell if the central 3x3 region covered by the 6-rate dilated kernel’s center weight belongs to a bus front or a train front when the surrounding areas are obscured (as a 3x3 non-dilated kernel perceives). The dilated kernel captures rich contextual information that the vehicle runs on railways and platforms are on both sides. With this information, this vehicle is much more likely to be classified as a train rather than a bus. In contrast, the dilated kernel in figure 1(c) is too aggressive and it reaches too far to capture helpful information. Areas 50m away has little to do with the class of the target building concerned. 12-rate and 18-rate dilated kernels are much more
aggressive and likely to capture irrelevant noise. Rather, dilated kernels with small rates that capture the boundary and adjacent buildings or roads may help. Thus, this structure loses its advantages in remote sensing images.

Although U-Net utilizes feature maps of all sizes and recovers the result in a stepwise fashion, it extracts surplus semantic information for our task and is hard to train. Besides, some feature maps to be concatenated in the decoder from earlier layers may be harmful. The goal of this paper is to demonstrate the importance of spatial information for the overall accuracy of our task and propose a method to deal with small and densely-distributed objects. Our model picks only useful feature maps from the encoder. In addition, large kernels make it shallow and easy to train without decreasing the receptive fields.

The main contributions of this paper are listed as follows:
- We modify popular state-of-the-art models in recent years to demonstrate the importance of spatial information in semantic segmentation of remote sensing images.
- A Stepwise-Refined Large-Kernel Deconvolutional Network is proposed to deal with small and densely distributed objects in semantic segmentation of remote sensing images.

2. Proposed Models
In this section, we present our model from encoder and decoder aspects respectively. Here we denote stride of feature maps as the ratio of the size of the original image to the size of these feature maps. For example, 4-stride feature maps indicate their size are 1/4 of the original input image.

2.1. Encoder
In image classification tasks, only high-level semantic features are desired. Thus, after the original image going through a serious of convolutional and pooling layers, feature maps are reduced to vectors and all spatial information are discarded. But in semantic segmentation, spatial information is vital to recover the segmentation mask Therefore, the encoder should preserve more spatial information at the expense of losing some semantic information. A balance has to be struck between the two kinds of information.

**Reduced VGG16**: FCN (with VGG16 as the encoder) reduces the size of feature maps to 1/32 of the original image’s size and produces a coarse segmentation mask, in which boundaries of objects are overly smoothed [3]. However, it becomes a fiasco on remote sensing images. Consider a house in figure 1(d), which covers less than 30 pixels. It will reduce to covering less than 1 pixel in the 32-stride feature maps. Furthermore, semantic richness in remote sensing images is much less than that in natural scene images. So we remove all layers after the 4th pooling layer and reduce the number all feature maps’ channels to 64. As a result, the model preserves only 3% parameters of VGG16 and produce 16-stride feature maps. We name it reduced VGG16.

**Dilated Resnet**: Dilated convolution is quite powerful in preserving spatial information and increasing receptive fields without increasing parameters. Deeplab v3 and PSPNet, take dilated convolution as an essential part. We use the same dilated Resnet as that in Deeplab v3.

2.2. Decoder
**Stepwise-Refined Large-Kernel Deconvolutional Network**: Compared to semantic segmentation of natural scene images, the effect of semantic information decreases a lot when spatial information is scarce in remote sensing images. So we design a Stepwise-Refined Large-Kernel Deconvolutional Network to focus on spatial information. This structure is composed of 4 blocks, with each block made up by a 2x upsample layer, followed by 64 7x7 convolutional kernels, followed by a batchnorm layer and a ReLU activation function. An overview of our model is shown in figure 2.

We recover the segmentation mask stepwise instead of in one step as FCN and Deeplab v3 do, considering that objects are too small and their boundaries are hard to be sharply recovered by a single upsample layer. In addition, large 7x7 convolutional layers is employed instead of their lightweight equivalent – 3 consecutive 3x3 convolution layers, as the latter deepen the network and take up more
memory to store feature maps, which in turn causes most GPUs of 12G memory running out of memory, including ours, despite the fact that the latter has less parameters. At last, 4-stride and 8-stride feature maps from the encoder are concatenated in the decoder to compensate for lost spatial information. 2-stride feature maps are ignored due to the large gap between them and feature maps in the decoder. Including 2-stride feature maps does harm to our model.

![Figure 2. Overview of our Stepwise-Refined Large-Kernel Deconvolutional Network.](image)

### 3. Experiments

#### 3.1. Dataset

The datasets in our experiments are ISPRS Vaihingen Challenge Dataset [9] and Fujian Dataset. Vaihingen Dataset consists of 3-band IRRG (Infrared, Red and Green) images. These images vary in size and average around $2500 \times 2000$. We choose 5 images (11, 15, 28, 30, 34) as test set and the other 11 images as training set, like most online submitter did.

Fujian Dataset contains 50 images captured by a satellite at 0.5m/pixel spatial resolution over a city in Fujian Province, China. The image sizes are $1000 \times 1000$. 10 classes are annotated and the colors for each class are specified in Table 1. White color are undefined regions and ignored when training. We conduct a 5-fold cross validation to evaluate our model on this dataset.

| Plantation  | Woodland | Lawn | Soil | Road |
|-------------|----------|------|------|------|
| 140,255,90  | 155,255,100 | 160,255,110 | 255,255,20 | 200,50,255 |
| Shanty town | Simple buildings | Roof | Facade | water |
| 92,92,92    | 200,200,200 | 229,229,229 | 180,180,180 | 100,200,250 |

#### 3.2. Configuration

Training policy, and parameter settings are detailed in the following paragraphs.

**Fine-Tune:** VGG16 and Resnet-101, the encoders in our model, are pre-trained on ImageNet. All decoders and the reduced VGG16 are trained from scratch.

**Data augmentation:** Due to the insufficiency of data, data augmentation is necessary to train a decent model. We augment our data in the following three ways. First, images are left-right flipped randomly with a probability of 0.5. Second, images are rescaled randomly from 0.5 to 2.0 times the original size. Third, crop a $513 \times 513$ patch from the image produced in the second phase. Before the third phase, an image might be smaller than $513 \times 513$ if shrunk too much in the second phase. In this case, some pixels should be padded to the right and bottom of the image, in order to guarantee it’s not smaller than $513 \times 513$. These pixels will be ignored when the loss is calculated as they are useless for training.

**Optimization:** Our models are implemented using Tensorflow and trained on a single NVIDIA TITAN Xp. We use SGD with batch size of 8, momentum of 0.9 and weight decay of 0.0005 to train our models. Each model is trained for 9000 epochs (Data augmentation substantially reduces the risk of overfitting if over-trained). Initial learning rates for VGG16 and Resnet-101 are set to 0.001 and
0.007 respectively. For each model, the learning rate is controlled by the polynomial learning rate policy:
\[ \alpha_k = \alpha \times \left(1 - \frac{k}{m}\right)^p \]
where \( \alpha_k \) is the learning rate at iteration \( k \), \( \alpha \) is the initial learning rate, \( k \) is the current iteration, \( m \) is the number of iterations to finish the training process and \( p \) is power mentioned above.

In our experiments, we report a common metrics from semantic segmentation and machine learning: mean Intersection over Union (mean IoU).

3.3. Results and Discussion

| Table 2. Results of models on the dataset. “+” separates the encoder and decoder. |
|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| Model           | mIoU V\(^a\)    | mIoU F          | Model           | mIoU V\(^a\)    | mIoU F          |
| VGG16+UP\(^b\)  (FCN)\(^c\) | 46.46%          | 47.15%          | Res+UP          | 55.11%          | 56.16%          |
| Reduced VGG16+UP| 46.88%          | 47.37%          | Res+SRLKD\(^d\) | 55.21%          | 56.47%          |
| Res+ASPP (Deeplab v3) | 54.68%          | 56.10%          | **Res+SRLKD-F8-F4** | **59.18%** | **58.86%** |
| Res+ASPP-F4\(^e\) (Deeplab v3+) | 58.69%          | 57.71%          | Res+SRLKD-F8-F4-F2 | 58.60%          | 57.95%          |

\(^a\) “mIoU V” and “mIoU F” stand for mean IoU on Vaihingen Dataset and Fujian Dataset respectively.
\(^b\) “UP” stands for a single upsample layer.
\(^c\) If a combination has been proposed before, we put its name inside the parentheses.
\(^d\) “Res” stands for Resnet101 which employs dilated convolution, producing 16-stride feature maps.
\(^e\) F8, F4 and F2 means 8-, 4- and 2-stride feature maps are concatenated in the decoder.
\(^f\) SRLKD stands for our proposed Stepwise-Refined Large-Kernel Deconvolutional Network.

Table 2 list the performance of previous state-of-the-art models and our proposed models.

The reduced VGG16, with only 3% parameters of FCN and trained from scratch, outperforms the original FCN by a small margin. It indicates that most semantic information carried by VGG16 is redundant, and it doesn’t benefit recovering the boundaries of objects in remote sensing images. In contrast, doubling the size of the last feature maps and exchanging more spatial information with semantic information helps improving the accuracy.

It is also worth noting that Res+UP outperforms Deeplab v3 by a little margin. Note that ASPP in Deeplab v3 is a superset of UP, but performs worse than that. This implies that the contextual information captured by ASPP is unnecessary or even harmful. Deeplab v3+ saves the situation purely by merging 4-stride feature maps based on Deeplab v3, and improves the performance obviously.

These aforementioned experiment results all demonstrate the insignificance of surplus semantic information and the importance of spatial information, which in turn encourages us to further focus on the spatial information. This is where our model comes in.

The best results are obtained by Res+SRLKD-F8-F4. SRLKD is designed to be quite shallow as it has no need to further extract semantic features. It recovers the segmentation mask in a stepwise fashion and boundaries of objects are much more refined. See figure 3(d). Besides, the main cause of the mixing of adjacent houses in the shanty town is the undefined white areas, as seen in figure 3(b). Further annotation are required to solve this problem. It’s noteworthy that with additional 2-stride features concatenated, the performance of Res+SRLKD-F8-F4-F2 drops by a small amount. The gap between 2-stride feature maps and high-level feature maps largely accounts for this. 2-stride feature maps only go through a single 7x7 convolutional layer and they differ little from the original images, which contain little semantic information. In this case, these low-level feature maps cannot be concatenated to high-level feature maps seamlessly. Thus, they are more likely to introduce noise and do harm to our model rather than compensate for spatial information.

4. Conclusion
In this paper, we demonstrate that the spatial information in remote sensing images is more important than that in natural scene images, as objects in remote sensing images are small and densely distributed. Based on this, we propose a Stepwise-Refined Large-Kernel Deconvolutional Network to focus on spatial information. During the deconvolution process, we employ a large-kernel but shallow deconvolutional structure to recover the segmentation mask stepwise instead of in one step like FCN and Deeplab v3 do. Our model recovers the objects’ boundaries more sharply and outperforms the previous ones in remote sensing images. In the future, we plan to simplify our network to produce 8-stride feature maps and further preserve the spatial information given the constraints of GPU memory.

![Image](a)

![Image](b)

![Image](c)

![Image](d)

**Figure 3.** Visual results on the remote sensing dataset. (a) is the original image. (b) is the ground truth label. (c) and (d) are predictions of FCN and Res+SRLKD-F8-F4 respectively.

**Reference**

[1] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. 2014 Sep 4.

[2] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 770-778).

[3] Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation. InProceedings of the IEEE conference on computer vision and pattern recognition 2015 (pp. 3431-3440).

[4] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions. arXiv preprint arXiv:1511.07122. 2015 Nov 23.

[5] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. InInternational Conference on Medical image computing and computer-assisted intervention 2015 Oct 5 (pp. 234-241). Springer, Cham.

[6] Chen LC, Papandreou G, Schroff F, Adam H. Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587. 2017 Jun 17.

[7] Zhao H, Shi J, Qi X, Wang X, Jia J. Pyramid scene parsing network. InProceedings of the IEEE conference on computer vision and pattern recognition 2017 (pp. 2881-2890).

[8] Chen LC, Zhu Y, Papandreou G, Schroff F, Adam H. Encoder-decoder with atrous separable convolution for semantic image segmentation. InProceedings of the European Conference on Computer Vision (ECCV) 2018 (pp. 801-818).

[9] Cramer M, The DGPF-test on digital airborne camera evaluation–overview and test design. Photogrammetrie – Fernerkundung – Geoinformation 2010 (pp. 73-82).