CS-433 Machine Learning Project 2: Road Segmentation

Xiaoqi Ma, Diyuan Wu, Shuo Wen
School of Computer and Communication Science, EPFL, Switzerland

***Abstract*—Road segmentation is the popular research topic in past decade. In this project, attention gates and cascade mode dilated convolution layers are used in UNet and LinkNet, which has improved the performance. Finally, LinkNet with cascade mode dilated convolution layers performed best and achieved 0.907 in F1 score and 0.950 in accuracy on test-set.**

I. Introduction

Road segmentation [1] is a hot topic and can be used in many areas such as GPS and map updating. Road segmentation can be divided into different tasks based on different perspectives of the observer, and two main perspectives are the satellite perspective and the driver perspective. In this task, we focus on the satellite perspective, that is, the top down view from the sky as shown in Figure 1.

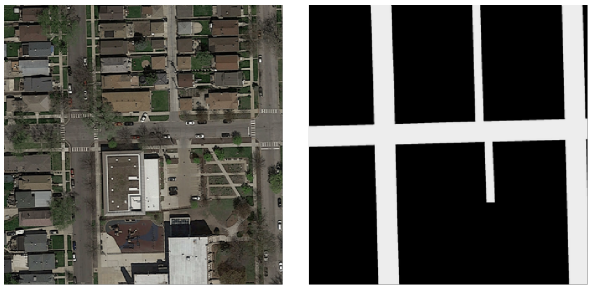

Figure 1: Example of input data and its label

From Figure 1 we can see that the main characteristics of the input images are as follows:

1) There are many occlusions in the image, such as trees blocking the road.
2) The road is thin and long and mostly spans the entire image.

Thus, according to those characteristics, we have the following challenges:

1) Considering only part of the picture can produce lots of mistakes because of the occlusions. For example, if we consider a part which should be the road but occlude by trees, it is impossible to predict it as road without the help of the other parts.
2) It is hard to perform localization with boundaries since the roads span the entire image.

To solve the two main challenges of this task, we build the model step by step. The most basic model is that first divides the images into many blocks, for example dividing a 400×400 image into 625 16×16 blocks. After that, using CNN[2], the model classify each block and then get the result of the whole picture. However, as we discussed in the first challenge, deciding whether a block belongs to road should take surrounding blocks into consideration. Thus, instead using small blocks as the input of the network, we use the entire image as the input.

Fortunately, there are already many existing models for performing segmentation on the image level instead of the block level, such as UNet[3], LinkNet[4], SegNet[5], and etc. However, these models are built for other types of datasets. For example, UNet is built for biomedical image processing and mostly it is used to segment the objects which are not as big as the entire image. According to the challenge 2, the road span the entire image and localization is almost impossible, so we should consider as many surrounding pixels as possible, that is, expanding the receptive field.

To expand the receptive field, we need to compress the length and width of the picture or increase the size of the convolution filter. Unfortunately, the input image size is 400×400 for trainset and 608×608 for testset, so after several times of pooling the image will become 25×25 or 19×19 and it will be difficult to continue doing regular pooling without trick. Also, pooling always cause varying degrees of information loss. Therefore, dilated convolution layers in cascade mode which enlarges the receptive fields are added to Linknet and UNet and get D-LinkNet[6] and D-UNet.

Another way to improve is to localize the road approximately before segmentation. Although structures which use boundaries to localization as F-RCNN [7]cannot be used, the attention gate can be introduced here. However, using dilated convolution layers with cascade mode and attention gates in the same network does not work in our experiment. Therefore, we add attention gates in UNet and LinkNet and get Attention-UNet[8] and Attention-LinkNet.

In this project, we choose UNet and LinkNet as our basic model and improve them by adding dilated convolution layers or attention gates. Finally, performs better and has achieved.

II. Method

In this section, the network architecture of the two models (D-LinkNet and Attention-LinkNet) will be introduced first. Then the two modules which bring improvement, that is, cascade mode dilated convolution layers and attention gate

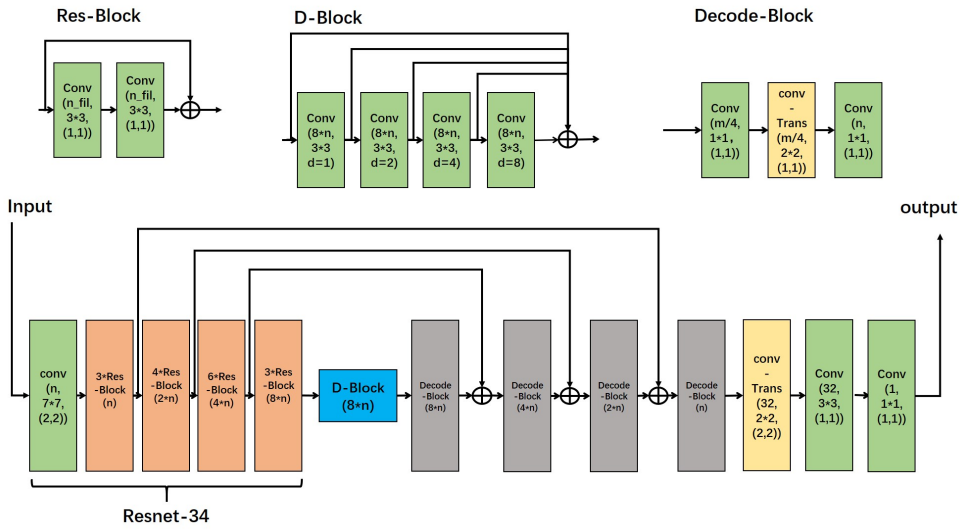

Figure 2: Architecture of D-LinkNet, n and m in the figure denote the number of filters

will be explained. Since the image sizes of trainset and testset of this task are 400×400 and 608×608, all the models are designed to receive the image whose width and height are multiples of 16.

A. Network Architecture

1) D-LinkNet: Adding dilated convolution layer into LinkNet, we get the final model D-LinkNet [6]. As shown in Figure 2, there are three parts in the network, named encoder part, dilated convolution part, and decode part. The original D-Linknet is used for images and masks of size 1024×1024 while in this project, the images and masks are of size 400×400. Thus, we slightly modified the original D-LinkNet to adapt this project. The modification of the original D-LinkNet is mainly two part: Firstly, we remove the pooling layer after the first convolution layer, the reason is that: each pooling layer will reduce the length and width of the training image to half, and 5 pooling layer will make the length and width $\frac{1}{32}$. Since the length and width of training image is 400 which is not a multiple of 32, we only use 4 pooling layer. Secondly, we change the output layer into the concatenation of a convolution layer, a convolution transpose layer, and a convolution layer.

The structure of our network is as follows: After a convolution layer with kernel size 7, stride 2 and 64 filters at the beginning, the input image will enter the encoder part. As shown in Figure 2, this part contains 4 encoders. The first encoder is a cascade of 3 Res-blocks and a pooling layer. Similarly, the second, third and fourth is a cascade of 4 Res-blocks and a pooling layer, 6 Res-blocks and a pooling layer, 3 Res-blocks and a pooling layer respectively. Since each encoder contain a pooling layer, thus the output image size will reduced to half after passing through each encoder. Finally, after this part, an input of size 400×400 will become a tensor of size 25×25 after the encoding blocks.

In the dilated convolution part, dilated convolution is stacked in the cascade mode as shown in Figure 2. There are 5 dilated dilated convolution layers, and the dilation rates of the layers are 1,2,4,8,16 (i.e. 2^i) . The reason for dilation rates selection will be described in Section II-B.

Corresponding to the 4 encoders in the encoder part, there are 4 decoders in the decoder part as shown in Figure 2. Each decoder is the concatenation of a convolution layer, a convolution transpose layer, and a convolution layer. The decoder part will output a tensor with the size of $400 \times 400 \times 64$. Finally, to get the output which is a binary image, we use one output block which contain the concatenation of a convolution layer, a convolution transpose layer, and a convolution layer. The main reason of the output layer is to change the output into a tensor of size 400×400.

2) Attention LinkNet: Similar to original LinkNet, Attention LinkNet also contains the same two parts: encoder part, and decode part, as shown in Figure 3. The main difference is that, in Attention LinkNet, we add an attention gate before adding the features of output of the encoder and the decoder. The structure of attention gate will be introduced in Section II-C.

The basic architecture is as follows: after the convolution layer at the beginning, the input images will enter the encoding part. The encoding part contains 3 encoder block, each encoder is the concatenation of 2 Res-block. Similarly, in the decode part, there are 3 decoder which is connected by the addition layer, and each decoder is the concatenation of a convolution layer, a convolution transpose layer and a convolution layer. The detailed structures of encoder and decoder are shown in Figure 4.

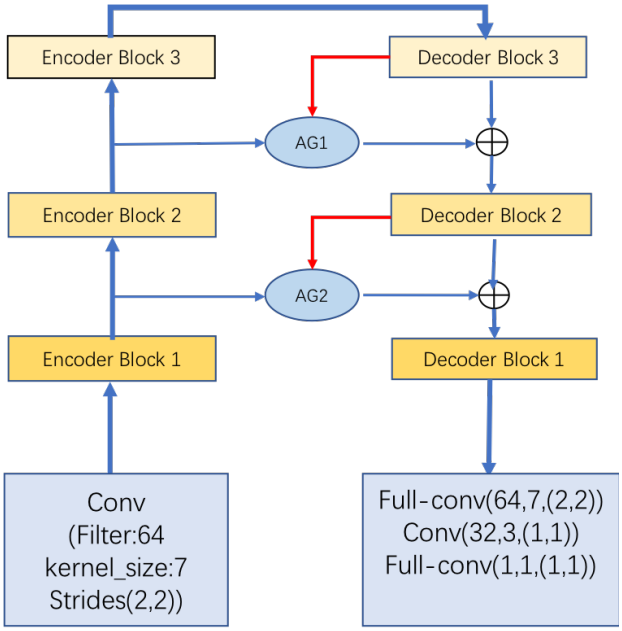

Figure 3: Architecture of Attention LinkNet

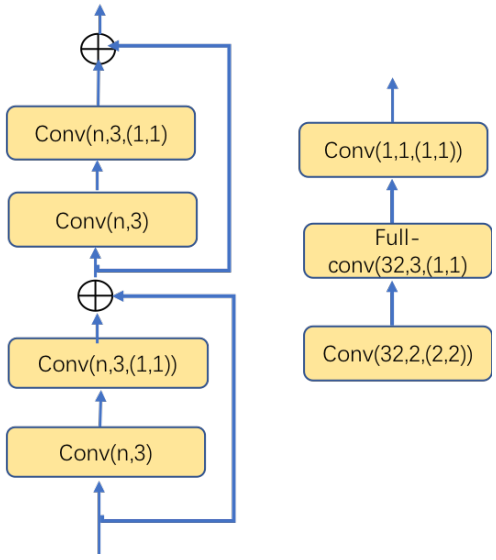

Figure 4: Encoder and Decoder in Attention LinkNet, the left is the encoder and the right is the decoder

B. Dilated Convolution Layer

The calculation rule of dilated convolution layer is shown in Figure 5. Instead of calculate the adjacent pixels, the filters calculate the spaced pixels to enlarge the receptive field. If the dilation rates of the dilated convolution layers are 1, 2, 4, 8 respectively, the receptive field of each layer will be 3, 7, 15,31. Since after 4 times encoder the size of the image will be 25×25 or 38×38, we choose the largest dilation rate to be 8 to fit both sizes.

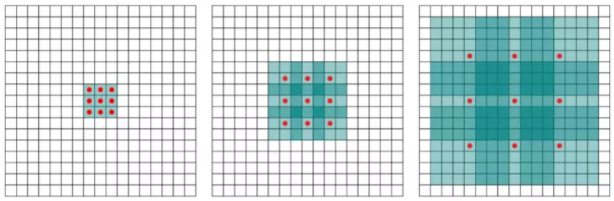

Figure 5: Calculation rule of dilated convolution layer[9]

C. Attention Gate

The encoder-decoder structure capture a sufficiently large receptive field through down sampling, but it is still difficult to predict the small objects, like the super thin road. To predict the small object, localization using boundary is widely used. However, localization with boundary does not work in this task since the road span the entire image. Thus, instead of using boundary to obtain the location, we use attention gate, which localizing with a heat map, to obtain the location. The structure of attention gate is shown in Figure 6.

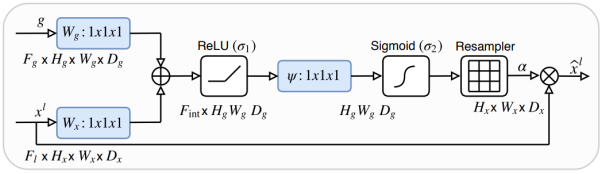

Figure 6: Structure of Attention Gate[8]

As shown in the figure, in the attention gate, we 2 different input x and g, where g called a gating vector. The propose of adding g is to determined the focus region of each pixel i.

III. Evaluation

A. Dataset and Data Augmentation

The dataset consist training images and test images, in the training dataset, there exist 100 original images and corresponding ground truth. In order to augment the training dataset, we first do the horizontal flip of images and ground truths at the same time. Then, the two hundred pictures are rotated by 90 degrees, 180 degrees, and 270 degrees separately and finally we get 800 images both in training images and ground truth.

Then we randomly select 20 percent of images and corresponding pictures from training dataset to form the validation dataset. The reason why we construct the validation data set is that when the loss of validation images is not changed, the callback function will adjust the learning rate to try to escape the local minimum. On the other hand, for the series of epoch, the h5file of weights will record the weight if validation loss of this epoch has declined,

For the training images, We perform three times 90-degree rotations on them and get 400 pictures, and then we flipped the 400 pictures horizontally, and finally got a training set of 800 pictures.

B. Implementation details

In the training phase, we use cross validation to find the best training epoch number. There are two ways of data separation. One is separate the data randomly first and then do the data augmentation; another one is first do the data augmentation and then randomly separate the data. However, the second method has a big problem: if we randomly separate the data after data augmentation, the validation data and the training data are highly correlated, so the loss of two parts will change consistently. Under this condition, overfitting cannot be found. Thus, we apply the first way, that is, separate the data randomly first and then do the data augmentation.

As for the loss, we use the dice-loss function based on dice coefficient, which is a measure of overlap of two samples. Compared with cross entropy which is just a proxy to maximize the back propagation, the dice coefficient directly maximize the metrics(F1-score) we need. The optimizer we used is the ADAM,which is a adaptive learning optimization algorithm.

C. Results

Roads (positive samples) in this dataset are only a small part, and the biggest part is background, that is, the recall rate is relatively more important in this task than the accuracy rate. Therefore, due to the unbalanced positive and negative sample sizes, both accuracy and f_1 score are used.

In this project, UNet, LinkNet, D-UNet, D-LinkNet, Attention-Unet, and Attention-LinkNet are implemented. The evaluation result of the models are shown in Table I.

Model	F1-score	Accuracy
UNet	0.891	0.939
LinkNet	0.896	0.945
D-UNet	0.902	0.948
D-LinkNet	0.907	0.954
Attention-UNet	0.899	0.946
Attention-LinkNet	0.900	0.947

Table I: Result

From Table I we can see that D-LinkNet performs best in this task, and dilated convolution layers bring more improvement than the attention gates.

D. Analysis

Two points are analysed in this section: the reason why the dilated convolution layers in cascade mode cannot work with attention gates and why dilated convolution layers bring more improvement than the attention gate.

First, there are two possible explanations about why the dilated convolution layers in cascade mode cannot work with attention gates. One possible explanation is dilated convolution layer in cascade mode combines many different levels of features to get the output, and such output cannot be used as gate vectors for attention gates since the feature level of gate vector and input vector should not differ too much. Another possible explanation is that the proper structure and parameters were not found.

Second, the reason why dilated convolution layers bring more improvement than the attention gate is that dilated convolution layers focus more on the overall information while the attention gates focus more on detailed information. In this task, the roads span the entire image and there are few detailed information in the input images, so the overall information is more important than detailed information.

IV. Conclusion

In this project, we implement 7 different neural networks: SegNet, UNet, D-UNet, Attention-UNet, Linknet, DLinknet, Attention-Linknet. We add cascade mode dilated convolution layers and attention gates to UNet and LinkNet. We find that both of the two methods are able to improve the performance and the dilated convolution layers works better. After comparison, D-LinkNet which gets the best result is chosen and it achieves 0.907 in f_1 score and 0.950 in accuracy.

REFERENCES

[1] R. H. Kallet, "How to write the methods section of a research paper," *Respiratory Care*, vol. 49, no. 10, pp. 1229–1232, 2004.

[2] L. O. Chua and T. Roska, "The cnn paradigm," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 40, no. 3, pp. 147–156, 1993.

[3] R. L. Barkau, "Unet, one-dimensional unsteady flow through a full network of open channels," *Computer Program, St. Louis, MO*, 1992.

[4] A. Chaurasia and E. Culurciello, "Linknet: Exploiting encoder representations for efficient semantic segmentation," in *2017 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2017, pp. 1–4.

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[6] L. Zhou, C. Zhang, and M. Wu, "D-linknet: Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction." in *CVPR Workshops*, 2018, pp. 182–186.

[7] X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep learning: An improved faster rcnn approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018.

[8] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N. Y. Hammerla, B. Kainz *et al.*, "Attention u-net: Learning where to look for the pancreas," *arXiv preprint arXiv:1804.03999*, 2018.

[9] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.