The Study of Skip Connections on Models for Scene Segmentation

Weitung Chen
weitung@mit.edu

Chenkai Mao
chenkai@mit.edu

Abstract

Scene Segmentation has always been a challenging problem in computer vision. While deep neural networks performs well in such problem, skip connections are often added to overcome the degradation issue. In this project, we examined the effects of adding different skip connections on models for scene segmentation evaluated on MIT ADE20K dataset. In particular, we tried single skip connection, mixing connections from different layers, and adding a 1x1 convolutional layer in the skip connections. The experimental data suggested that concatenating one skip connection from block 3 of the Resnet-50 encoder to the output of the decoder resulted in a better accuracy than other skip connection types. Overall, the single connection from block 3 can help enhance the accuracy of the model by around 0.7% in average.

1. Introduction

With its broad applications in autonomous driving[2], indoor navigation, and augmented reality systems, scene segmentation has becoming one of the key problems in computer vision and deep learning society. Compared with simple image captioning and classification problems, scene segmentation tasks require extra precision as the system needs to label the image pixel-wise. While it is a challenging problem, it also provides us more insight on understanding vision.

Throughout the year, various models have been proposed and implemented, like AlexNet[5], VGG-16[8], GoogLeNet[9], and ResNet[4]. While the model gets more and more efficient, the network is also getting deeper and more complex at the same time. In general, deep architectures suffered from degradation problem. Namely, when the network gets deeper, its accuracy becomes saturated and then degrades rapidly. Different methods have been applied to resolve this issue. Introducing skip connections is an approach commonly used in various models[7, 3].

Despite the outstanding performance of the encoder-decoder architecture with Resnet, there is still room for improvement, especially clearly recognizing the contour of an object. We observed that with the original encoder-decoder architecture, the decoder generated the output image from the compressed feature map of encoder, which is low dimensional and may lose some high frequency details. In this paper we aim to improve the model's ability to classify high frequency details by adding skip connections from early layers of encoder directly to decoder. We focus on examining different aspects of skip connections, including skip length, symmetry, mixing layers, having an additional 1x1 convolutional layer in the skip path etc. We trained our models based on MIT ADE20k data set[10, 11], under encoder model Resnet50Dilated and decoder model PPM-Bilinear-Deepsup as well as C1-Bilinear. By trying out different skip connections, we found that adding one skip connection resulted in a better performance than connecting multiple layers of output to the decoder. In addition, we observed that adding an additional 1x1 convolutional layer in the skip connections did not help enhance the performance of our models.

The remainder of this paper is organized as follow. In section 2, we reviewed the related work in the field of scene segmentation and properties of skip connections. Next, our different approaches and methods will be introduced. Finally, the experiment results will be presented in section 4 and the conclusion is summarized in section 5.

2. Related work

One popular architecture, originated in Recurrent neural network, is the encoder-decoder architecture[1]. According to this architecture, the encoder network transforms the input into a compressed representational feature map or tensor, and then the decoder takes in the representation and expands it to the final output. It has been shown that this framework has an outstanding performance in scene segmentation problem[10, 12].

Despite the astonishing performance of deep neural networks, they generally suffer from the degradation problem, which implies that the performance drops when the network becomes too deep. Various models were introduced to mitigate this problem. Residual network[4], introduced by Microsoft in 2015, addressed this problem by adding a forward residual connection as illustrated in Figure 2. The residual connection increases the stability of the network and thus provides a reliable solution for training large and deep networks.

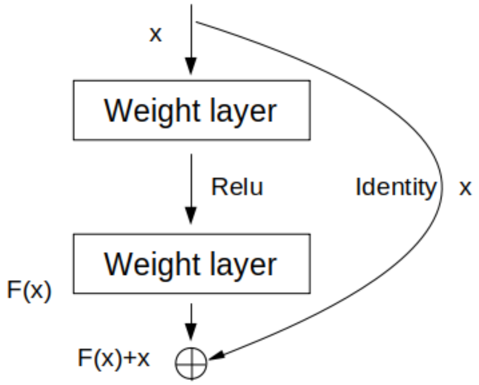


Figure 1. Building block of Resnet, with the identity residual connection.

The residual connection in Resnet falls under a more general category named skip connections, a commonly adopted technique in neural network models. There are various models that took advantages of skip connections[7, 6, 4], but there isn't a paper dedicated to explain and examine the different effects of skip connections on neural network models. Skip connections aren't limited to identity summation. In this paper, we would like to explore its possibility to skip multiple layers, and adding additional neural networks along the path.

3. Approach/Algorithm

We started with the basic encoder-decoder model with Resnet50-dilated as encoder, and we tried out two different decoder models, ppm-bilinear-deepsup and C1-bilinear. Then, we added different skip connections and tested out their performance. The architecture and the notion of skip connection are illustrated in Figure 2, which describes the model implementing Method 5 in our approach. Each model was trained on MIT ADE20K data set with 20 epoch-iterations and 10 epochs, alongside with a control model which is the original model without skip connection. The result is shown in section 4. Here we list different aspects we considered below, which are used in different approaches listed in section 4.

3.1. One Skip Connection

We started by adding one skip connection from different blocks of the encoder to different spots of the decoder. Detailed implementation for each of the methods are listed below:

- *Method 1*: Concatenate the block 3 of the encoder to the output of the pyramid pooling module
- *Method 7*: Concatenate the block 3 of the encoder to the output of the PPM-Bilinear-Deepsup decoder, which is the stage after applying convolution operation on the output of the pyramid pooling module. To obtain the correct number of classes at the output, we also added a convolution layer after concatenating the connection.
- *Method 8*: Add block3 to first layer of C1-Bilinear decoder
- *Method 9*: Add block3 to second layer of C1-Bilinear decoder
- *Method 12*: Add block4 to first layer of C1-Bilinear decoder
- *Method 13*: Add block4 to second layer of C1-Bilinear decoder

3.2. Multiple Connections from Different Layers

We also tested out the effects of multiple connections. Detailed implementation for each of the methods are listed below:

- *Method 2*: Concatenate the block 1, 2, 3, and 4 of the encoder to the output of the pyramid pooling module.
- *Method 3*: Concatenate the block 3, 4, and the three convolution layers before block 1 to the output of the pyramid pooling module.
- *Method 10*: Add both block 2 and 3 to first layer of C1-Bilinear decoder
- *Method 11*: Add both block 2 and 3 to second layer of C1-Bilinear decoder

3.3. Convolution Layer in Skip Connection

In this part, we investigated the effect of 1x1 convolutional layer on skip connections. Detailed implementation for each of the methods are listed below:

- *Method 4*: The connections and merging method is identical to method 2 but with 1x1 convolution layers in the connections.
- *Method 5*: The connections and merging method is identical to method 1 but with 1x1 convolution layers in the connection. An illustration of this model can be seen in 2.
- *Method 6*: The connections and merging method is identical to method 7 but with 1x1 convolution layers in the connection.
- All the skip connections in Table 2 used a 3x3 convolution layer to make sure the number of channels were aligned with the decoder.

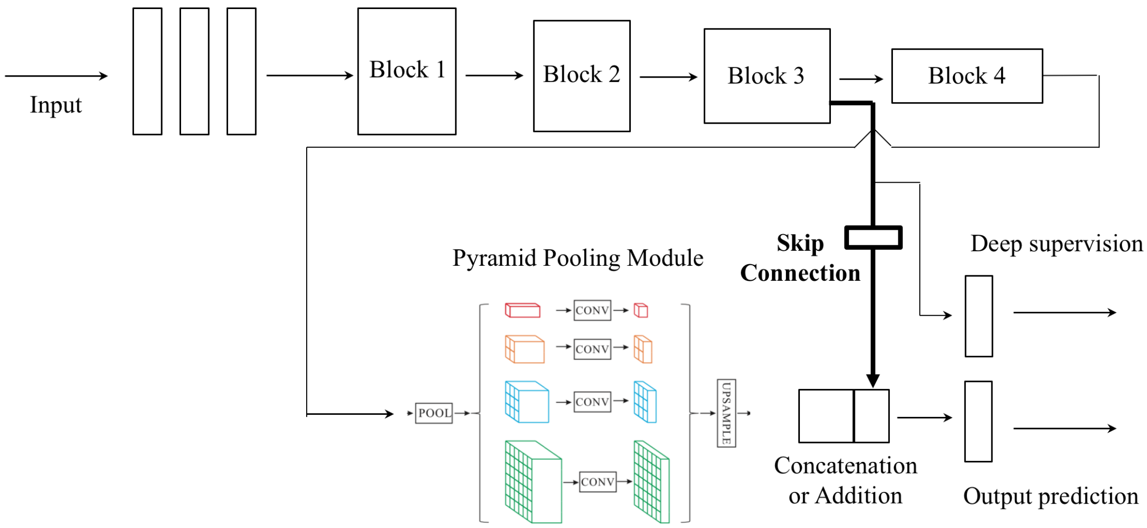


Figure 2. Encoder-decoder architecture with skip connections

4. Experimental Results

Based on the approaches introduced in section 2, we trained the model with the ADE20K dataset. The performance of our models is evaluated in terms of segmentation accuracy (%) on the validation dataset. Comparison of visual segmentation results using the original controlled model and our best performed model with one skip connection are also displayed in Figure 3.

4.1. Validation Accuracy

Evaluation of validation dataset is shown in Table 1 and Table 2. Each model was trained with different methods and used the parameters defined in the caption. Due to time constraint on this project, we only trained the model for 10 epochs with 20 iterations per epoch. However, to ensure the accuracy value was reliable, we trained some key models multiple times to validate that the statistical fluctuations is small compared to our model differences.

Comparing method 1,2,and 3 in Table 1, we are able to observe that a single skip connection on the model resulted in a better accuracy than mixing multiple connections. This observation can also be supported by comparing the methods in Table 2. For the models in which we used the pyramid pooling module as the decoder, the validation accuracy were even slightly greater than the original model with no skip connection. On the other hand, the 1x1 convolution layers which we added on the skip connections didn't seem to enhance the overall performance of the model, and it even decreased the validation accuracy in some cases (compare methods 2,3,and 4 in Table 1). This situation was even worse in Table 2), where all the methods have used 3 by 3 convolution layers.

Method	Accuracy (%)
Control - no skip connections (*)	63.09
1 - single connection from block 3 (*)	63.87
2 - multiple connections block 1-4 (*)	63.29
3 - multiple mixed connections	62.78
4 - multiple with 1x1 convolution	61.73
5 - single with 1x1 convolution	63.41
6 - single after ppm with 1x1 convolution	63.99
7 - single after ppm (*)	63.75

Table 1. Validation Accuracy obtained when applying the methods on the encoder:Resnet50-dilated8 and decoder:ppm-bilinear-deepsup, concatenating skip connections to the output of the decoder. Other parameters were set to epoch=10, iterations=20, optimizers=SGD, learning rate(lr)=0.02, lr decay=0.9, momentum=0.9, weight decay=0.0001, deep supervision scale=0.4
(*) implies the accuracy value was obtained by training the model multiple times and used the average percentage

Although different decoder was used in two tables, it is the comparison with the control that we observed that concatenating skip connections to the output of the decoder is a better mean than summing them. Thus, according to the observations above, we chose a model with one skip connection, no convolution layer in the connection, and use concatenation technique to append the connection to the output of the decoder, to visualize the segmentation result and compare it with the original controlled model without skip connections in section 4.2.

Method	Accuracy (%)
Control - no skip connections (*)	62.01
8 - single connection from block 3	60.15
9 - single from block 3 (longer skip)	60.72
10 - multiple mixed connections	55.41
11 - mixed connections (longer skip)	54.4
12 - single connection from block 4	56.83
13 - single from block 4 (longer skip)	58.87

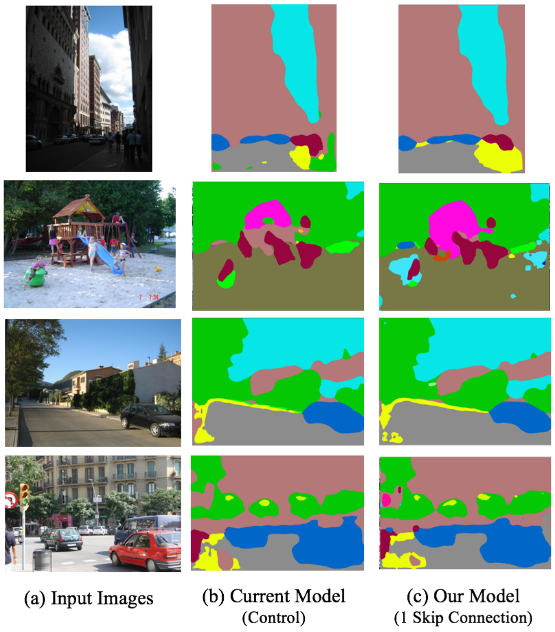
Table 2. Validation Accuracy obtained when applying the methods on the encoder:Resnet50-dilated8 and decoder:C1-bilinear, summing skip connections with the output of the decoder. All other parameters were set to epoch=10, iterations=20, optimizers=SGD, learning rate(lr)=0.02, lr decay=0.9, momentum=0.9, weight decay=0.0001, deep supervision scale=0.4.
(*) implies the accuracy value was obtained by training the model multiple times and used the average percentage

4.2. Segmentation Results

We used the model with best performance found in section 4.1 and test it with random 4 images in the validation dataset. The results are shown in Figure 3. We can see that while both results are mostly similar in terms of the labels predicted, our model made some correct predictions that the original model didn't make. An example of this will be the top row of Figure 3. The original controlled model suggested that there will be a tree (color green) at the bottom right corner of the image whereas the input image told us there aren't trees in that corner. However, our model made the correct prediction and gave that portion of the image a label of sidewalk (color yellow). From the validation accuracy and the visualized segmentation results, we can see that our model with one skip connection has slightly improved the accuracy of scene segmentation process.

5. Conclusions

We have investigated different skip connections between encoder and decoder of the scene segmentation network. We found that adding one skip connection resulted in a better performance than connecting multiple layers of output to the decoder. Also, we can conclude based on the overall performance difference difference of two tables, concatenating the skip connections is a better mean than summing them to the decoder of the model. According to the experimental results, we observed that adding 1x1 or 3x3 convolutional layer on the skip connections didn't help enhance the overall performance of the scene segmentation model. In the future, we aimed to train our models with more epochs and iterations in the hope to examine the performance of skip connections on a more stable model.


(a) Input Images

(b) Current Model (Control)

(c) Our Model (1 Skip Connection)

Figure 3. Visual Comparison of Segmentation Results using the Original Controlled Model and Our Best Performed Model (Method 1)

6. Personal Contribution

In this project, we brainstormed and worked together almost the whole time and splited all the work in half. Each of us worked on half of the presentation slides, the final report, and a different approach in the study. Detailed explanation of the personal contribution are listed as follow.

6.1. Weitung Chen

Throughout the project, I tried several skip connection methods on the model with encoder Resnet50-Dilated8 and decoder PPM-Bilinear-Deepsup and obtained all the data in Figure 1. I was also responsible for keeping the experimental data as well as creating graphs and figures from them.

6.2. Chenkai Mao

In this project, I came out with the skip connection idea and developed it together with Weitung. I tried different skip connection methods on the model with encoder Resnet50-Dilated8 and decoder C1-bilinear. Different from Weitung, I used addition when creating the skip connections. The results are shown in Table 2.

References

[1] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[2] A. Ess, T. Müller, H. Grabner, and L. J. Van Gool. Segmentation-based urban traffic scene understanding. In *BMVC*, volume 1, page 2. Citeseer, 2009.

[3] A. Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[6] X. Mao, C. Shen, and Y.-B. Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in neural information processing systems*, pages 2802–2810, 2016.

[7] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[8] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[10] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[11] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, pages 1–20, 2016.

[12] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, and A. Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal on Computer Vision*, 2018.