Review: VGGNet

Jiho Bak
School of Electrical and Electronic Engineering, Yonsei University

Abstract

The main concept of VGGNet [4] is clear, making a deeper convolutional network with small filters. K. Simonyan et al. [4] claims that these networks show the state-of-the-art performance in ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) (at that time), even using conventional ConvNet architectures [5], and less(or same) parameters. In the paper, the advantages of depth with 3×3 conv. layers with stride one, were elaborated in both theoretical and experimental perspectives. The experiment was done with 11-19 depth layers under precise conditions, specially in ImageNet, and also in other datasets (e.g. Caltech, VOC) to show the generalization performance. To add my view, this paper introduces the importance of the depth in convolutional networks, and shows that those deep depth can be actually achieved with simple 3×3 conv. layers.

1 Why Deep 3×3 Convolution?

Unlike other top-performing models, VGGNet [4] does not use large receptive fields in first and other layers. It uses multiple 3×3 conv. layers instead, covering similar receptive fields. For example, three 3×3 conv. layers has an receptive field of 7×7. There are two advantage of this structure. First, since one layer is replaced to several layers and each layers got activation functions(ReLU), non-linearlity of model increases. The increased non-linearlity makes the decision funciton more discriminative [4]. Second, the number of parameters decreases. The simple calculation with fixed number of channels is done in the paper. One 7×7 conv. filter uses 81% more parameters than of three 3×3 conv. filters.

The previous attempt of using small conv. filter exists [1] but it's less deeper and not evaluated in large-scale dataset like ILSVRC. As a result, since top-performing models got deeper network, the performance of the deep 3×3 conv. layers are tested in the paper.

2 VGGNet Configuration

The experiment was set to test the performance of suggested concept, so models follow the conventional/simple architecture over all. The layer structures are organized in the table 1. The stride and padding is fixed to 1 in all conv3 layer, which preserves the spatial resolution. The maxpool layer uses 2×2 window of stride 2. All hidden layers is followed by ReLU function. The LRN normalization was not applied from model B since the performance was good enough without LRN while testing A.
The author also explains that 1×1 conv1 layer increases non-linearlity of decision function without affecting the receptive fields which was utilized by Lin et al. [3].

3 Classification Framework

3.1 Train

Loss Function and Optimizer

Multinomial logistic regression objective(which is cross-entropy loss) function was used for loss function. The mini-batch gradient descent with momentum was applied as optimizer with batchsize of 256, momentum of 0.9 and scheduled learning rate with the initial value of 10^{-2}. For regularization, dropout ratio 0.5 is applied for first two fc layers and L2 penalty multiplier of $5 * 10^{-4}$ was applied to loss function.

Pre-initialization

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

Figure 1: ConvNet configurations

The parameters of middle conv. layers(fifth to last), from trained model A, were used for pre-initialization of other deeper networks. Other initialization was done by sampling from a normal distribution, since the author noticed Xavier Initialization [2] after submission.

Input Image Size

Basically, the input size of model is 224×224. So if rescaled image is bigger, the image is cropped before input. The model is trained in two different ways, single-scale(256, 384) and multi-scale([256:512]).

3.2 Test: Dense Evaluation and Multi-crop Evaluation

Dense evaluation method evaluates the model with arbitrary size of input. This can be done by converting fully-connected layer into 1x1 conv. layer and making the network fully-convolutional. Dense evaluation is mainly used in this paper. But additionally, since the author claimed that dense and multi-crop evaluation methods are complementary, multi-crop evaluation and the mixture of them was also tested.

4 Experiment Results and Conclusion

The top-1 and top-5 errors were measured under the conditions set in train and test steps. Single-scale 'testset' evaluation, multi-scale 'testset' evaluation and multi-crop evaluation were experimented, and finally the ensemble model was tested.
Overall, deeper model got lower error. To compare single-scale and multi-scale evaluation, the error of multi-scale evaluation was lower. For multi-crop evaluation, multi-crop evaluation error was lower but the mixture of dense and multi-crop evaluation method was the lowest. This result supports the author's claim that two methods are complementary. As expected, the ensemble of model D, E performed best. Comparing with GoogLeNet, it performed better than single model, and almost closed to 7-ensemble model(VGG error(2 nets): 6.8%, GoogLeNet(7 nets) error: 6.7%).
As a result, since VGGNet performed almost as sota models just with conventional architecture, and also the generalization performance was confirmed on appendix, we can say that the effectiveness of deep 3×3 conv. layer is proved.

Reference

[1] Jonathan Masci Luca M. Gambardella Jurgen Schmidhuber Dan C. Cires¸an, Ueli Meier. Flexible, high performance convolutional neural networks for image classification. *IJCAI*.

[2] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. *AISTATS*.

[3] Chen Q. Lin, M. and S. Yan. Network in network. *ICLR*.

[4] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *ICLR*.

[5] J. S. Denker D. Henderson R. E. Howard W. Hubbard Y. LeCun, B. Boser and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*.