Deep Convolutional Neural Networks for Large-Scale Image Classification

Xinwei Wang and Jue Wang

Neural Networks/Deep Learning - COGS 181

Prof. Zhuowen Tu

25 March 2017

Deep Convolutional Neural Networks for Large-Scaled Image Classification

Abstract

Deep convolutional neural networks have shown great potential in recent years in the field of computer vision. However, they may still result in problems such as difficulty of training and overfitting. In this paper, we explore how does depth of a certain type of convolutional neural network effect the classification performance on a large-scale class with small class size. We also compare the performance of the networks with optimal depth across different neural network architectures, such as ResNet and VGG network, to show if extra layers indeed improve the performance of convolutional neural networks in general. The results of experiments reflect that deep convolutional neural networks perform better than shallow ones in general, and deep VGG networks are more powerful than deep ResNets.

Introduction

Currently convolutional neural network has become a popular tool in solving classification problems, such as MNIST digit-recognition, video analysis, language processing, and many other fields (Cires¸an, et al., 2012). As the dataset become more challenging, large deep convolutional neural networks are trained to fit the data into a more complexed classification model (Krizhevsky, et al., 2012). It achieves great success recently especially when Karen Simonyan and his team secured the first place in ImageNet Challenge 2014 by training a deep convolutional neural network, and proposed the basis of their idea in 2015 (Simonyan and Zisserman, 2015).

As the model become more complicated, overfitting could be one of the serious problems that will damage the performance of such neural networks. Some techniques such dropout were developed to address this problem (Srivastava, et al., 2014). However, it is worthy questioning that even with those adjustments, are deep network structures indeed more powerful than shallow network structures for all kinds of convolutional neural network architectures? And does deep network can be applied to improve the classification accuracy for all kinds of data?

We propose to compare how well does the Visual Geometry Group neural network and residual network with various numbers of layer perform on classifying a labeled dataset with small number of classes. From the comparison across different convolutional neural network structures, we can further explore whether the deep convolutional neural network in general will be more powerful than shallow convolutional neural networks on different kinds of dataset.

Data

To investigate these problems, we will only take consideration of large-scale dataset with small number of classes, and evaluate the performances of different types of convolutional neural networks with different depths on this specific dataset. The motivation is that as Simonyan's team proposed deep convolutional neural network and confirmed its power of classifying large-scaled dataset with large number of classes (ImageNet dataset), we are looking for drawing a comparison with it. Thus we choose to train the models on CIFAR-10.

The CIFAR-10 dataset is a labeled subset containing the 80 million tiny image datasets collected by Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The dataset consists of 60000 32x32 colorful images from 10 classes, with 6000 images per class. There are 50000 training images and 10000 test images. The test set contains exactly 1000 randomly-selected images from each class. The training set contains exactly 5000 images from each class. The training set is divided into five training batches, each with 10000 images with random order, but some training batches may contain more images from one class than another.

Method

Residual Network(ResNet)

The general idea of residual network is as follows: let be a convolution or a series convolution of x, merge it with x by addition, and apply rectified linear unit activation function to it. The central idea behind is that ResNet learn to correct the residual errors. Once the representation learned performs well, the extra layers should wrap the representation too much.

We are going to adapt the residual network proposed in Deep Residual Learning for Image Recognition (He, et al., 2016). The residual network is composed of units and residuals. each units contains certain number of residuals (which is up to us when choosing the total number of layers), and each residual contains 2 sublayers. There are 3 units in total and fully connected layers after 3 units. We choose 3 depth for our experiment: 5 residuals per unit, 18 residuals per unit, and 27 residuals per unit, which is equal to, 32 layers, 110 layers, and 164 layers residual network.

Visual Geometry Group Neural Network(VGG)

VGG is the ConvNet architecture developed by Visual Geometry Group of University of Oxford, which not only achieve the state-of-the-art accuracy on ILSVRC classification, but are also applicable to other image recognition datasets. VGG uses a stack of 3x3 convolutional layers and 1x1 convolution filters. Then a Max-pooling is applied to each 2x2 window with stride 2. The stack of convolutional layers is then followed by three fully connected layers, with the last one as soft-max layer that returns the class. Each hidden layer will be equipped by rectified linear unit activation function (Simonyan and Zisserman, 2015).

In VGG11, we adapt a stack of two 3x3 convolutional layers with kernel size with 64 inputs and 64 outputs (unit 1), two 3x3 convolutional layers with 128 inputs and 128 outputs (unit 2), and four 3x3 convolutional layers with kernel size with 256 inputs and 256 outputs(unit 3). With three fully connected layers, we have 11 layers in total. In VGG16, we adapt a stack of two 3x3 convolutional layers with kernel size with 64 inputs and 64 outputs (unit 1), two 3x3 convolutional layers with kernel size with 128 inputs and 128 outputs(unit 2), three 3x3 convolutional layers with kernel size with 256 inputs and 256 outputs(unit 3), and six 3x3 convolutional layers with kernel size with 512 inputs and 512 outputs(unit 4 and 5). With three fully connected layers, we have 16 layers in total. In VGG19, we adapt a stack of two 3x3 convolutional layers with kernel size with 64 inputs and 64 outputs(unit 1), two 3x3 convolutional layers with kernel size with 128 inputs and 128 outputs(unit 2), four 3x3 convolutional layers with kernel size with 256 inputs and 256 outputs (unit 3), and eight 3x3 convolutional layers with kernel size with 512 inputs and 512 outputs(unit 4 and 5). With three fully connected layers, we have 19 layers in total. All the above input and output number ignore the origin input, final output, and the conversion between different units.

Experiment

There are two variables in our experiment settings: the convolutional neural network architectures and the number of layers. Hence control variate method will be adapted to test whether each variable influences the classification performance of the neural network. Following this method, one variable will be fixed at first. Thus, we look for the optimal number of layers within each convolutional neural network at first, and then compare how well does VGG and ResNet perform with their optimal depths.

However, directly choosing the parameter corresponding with the highest testing accuracy will underestimate the true classification error of the model. Hence, we will adapt training-validation method to solve the problem. The training batch will be partitioned to two sets, with proportion of 20% and 80%. The smaller set is validation set, and larger one is training set. Each time the convolutional neural network will be trained to optimize the training accuracy, and then we choose the number of layers with highest validation accuracy to be the optimal value of that parameter.

At last, the true classification accuracy of each convolutional neural network with the optimal number of layers is approximated by the correct prediction percent on the test set. The true accuracies will be compared across the two types of convolutional neural networks to decide whether the deep neural networks provide better classification accuracy than shallow neural network does.

Since ResNet and VGG network have very different structures, it is impossible to test the performance of the same number of layers within each architecture. Thus, we perform tests on reasonable numbers of layers, which are relatively deep or relatively shallow within each architecture. We choose 3 depths for our experiment of ResNet: 5 residuals per unit, 18 residuals per unit, and 27 residuals per unit, which is equal to, 32 layers, 110 layers, and 164 layers residual network. We adapt the VGG structure and choose 3 types of depth for visual geometry group neural network: VGG11, VGG16, and VGG19.

As the training set is randomly portioned to 5 proportions, we fix one of them to be validation set and the remaining to be training set. Both neural network architectures will be trained with an increasing order of number of layers. We use cross-entropy loss and classification accuracy to evaluate our models.Each time the images of loss and training accuracy of the trained model will be plotted, and a validation accuracy will be returned.

During the training of ResNet, we divide data into batches with batch_size 128. We train our residual network for a total of 10k iterations, using learning rate 0.1 and weight decay rate 0.0002. In addition, we adapt leaky ReLu as our activation function with ReLu leakiness 0.1.

During the training of VGG, we divide data into batches with batch_size 128. We use 0.1 as our learning rate and 0.0001 as our decay rate, max pooling after each unit with dropout rate 0.25, and ReLu as activation function after fully connected layers with dropout rate 0.5.

After the optimal number of layers is chosen based on validation accuracy, the ResNet and VGG with that specific depth will be evaluated and compared with each other based on their prediction accuracy on the testing batch.

Experiment Result

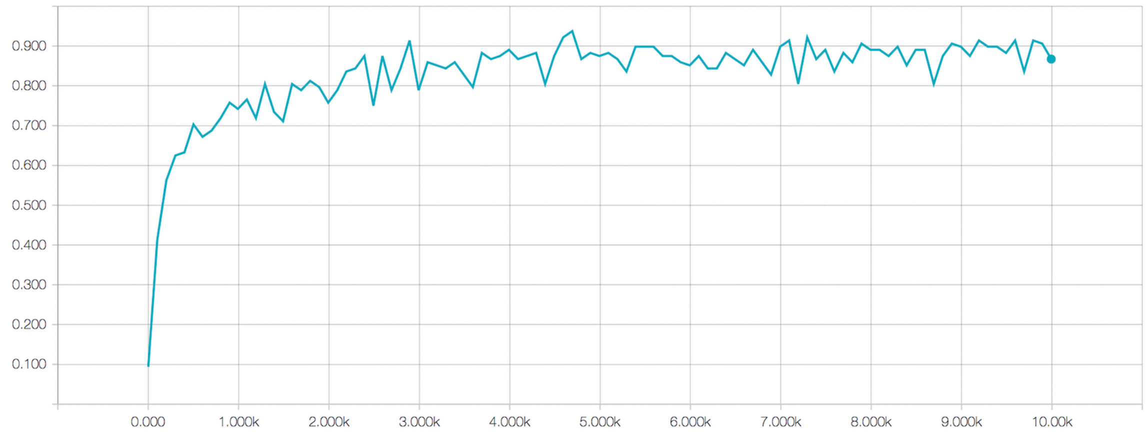

Figure 1: Training accuracy of Residual network with 32 layers. **Maximum Accuracy: 0.9375**

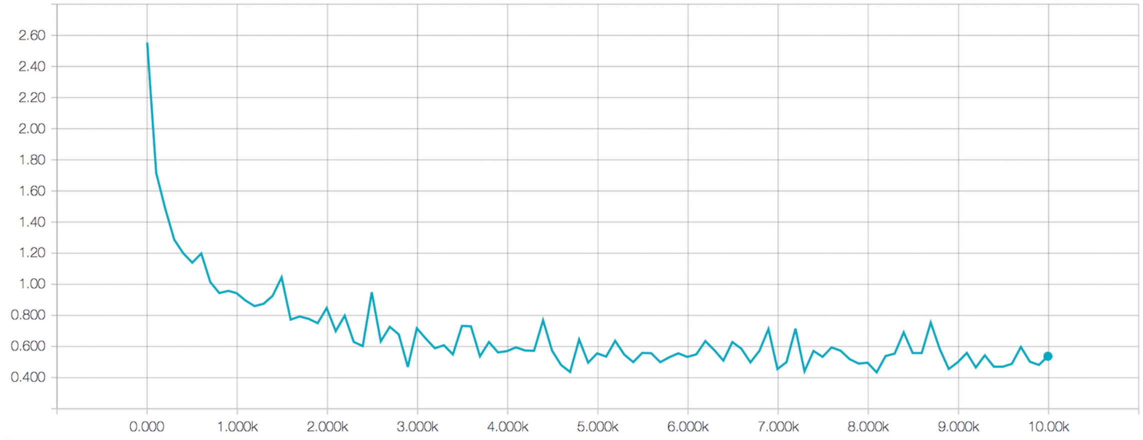

Figure 2: Training loss of Residual network with 32 layers. **Minimum Loss: 0.4360**

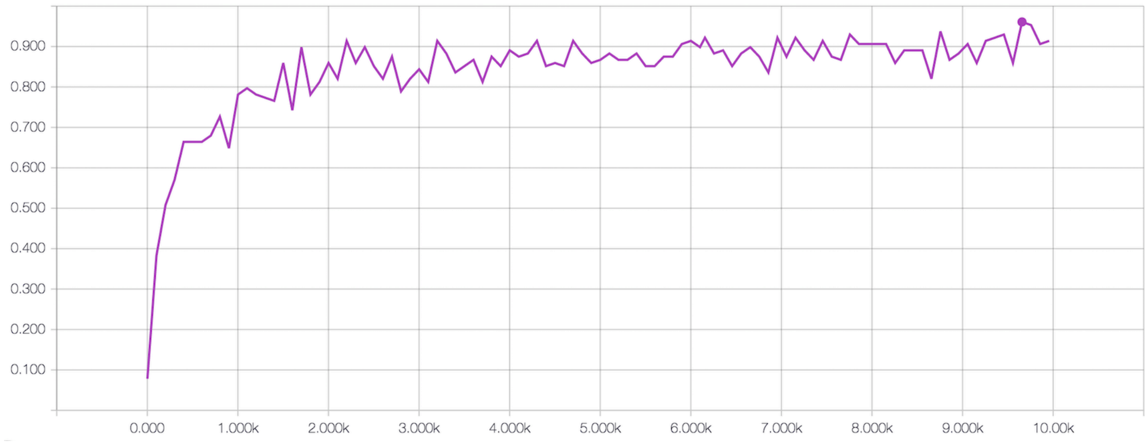

Figure 3: Training accuracy of Residual network with 110 layers. **Best Accuracy: 0.9609**

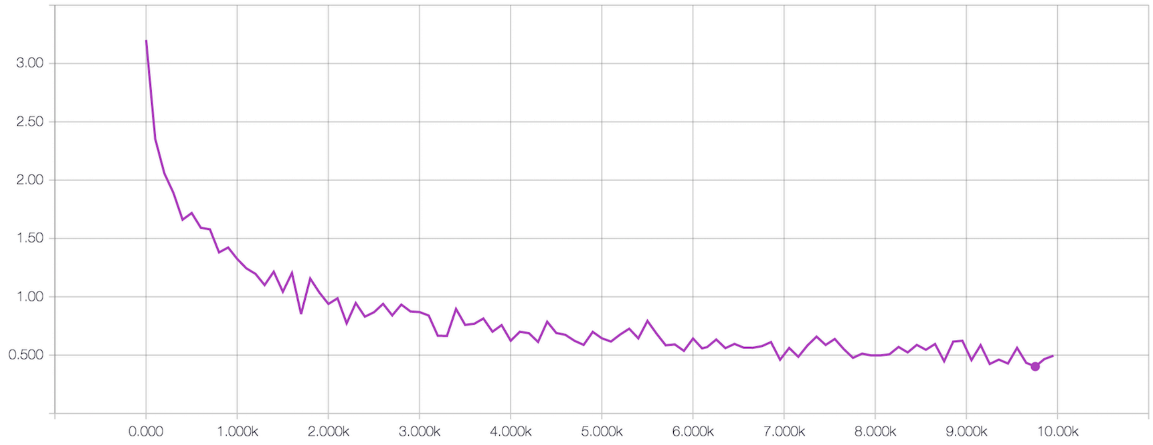

Figure 4: Training loss of Residual network with 110 layers. **Minimum Loss: 0.4020**

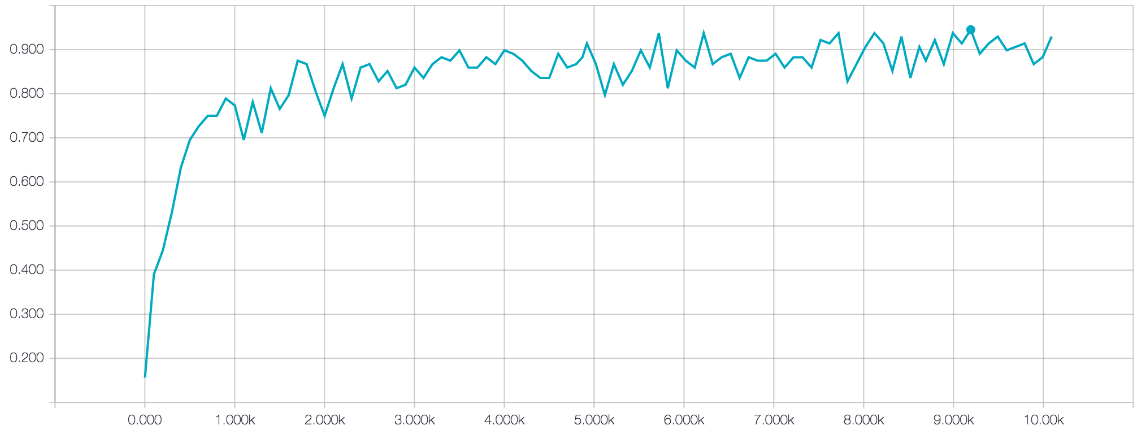

Figure 5: Training accuracy of Residual network with 164 layers. **Best Accuracy: 0.9453**

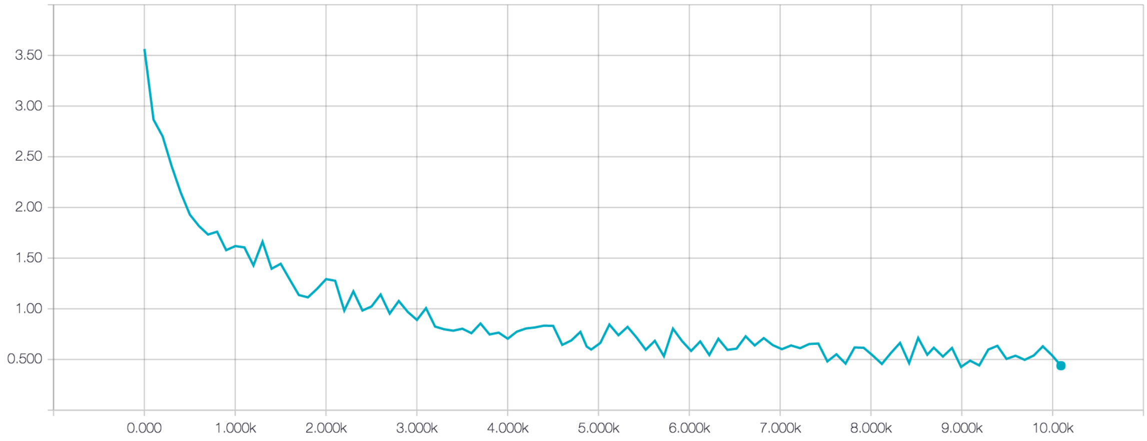

Figure 6: Training accuracy of Residual network with 164 layers. **Minimum Loss: 0.4266**

Number of Layers	Validation Accuracy	Test Accuracy
32	0.854	N/A
110	0.856	N/A
164	0.906	0.898

Table 1: Validation Accuracy of Residual Networks and the Test Accuracy for the optimal structure among three experimenting structure.

Table 1 reflects that as the residual network gets deeper, it achieves higher accuracy. Thus, 164 is the optimal number of layers chosen from the training process.

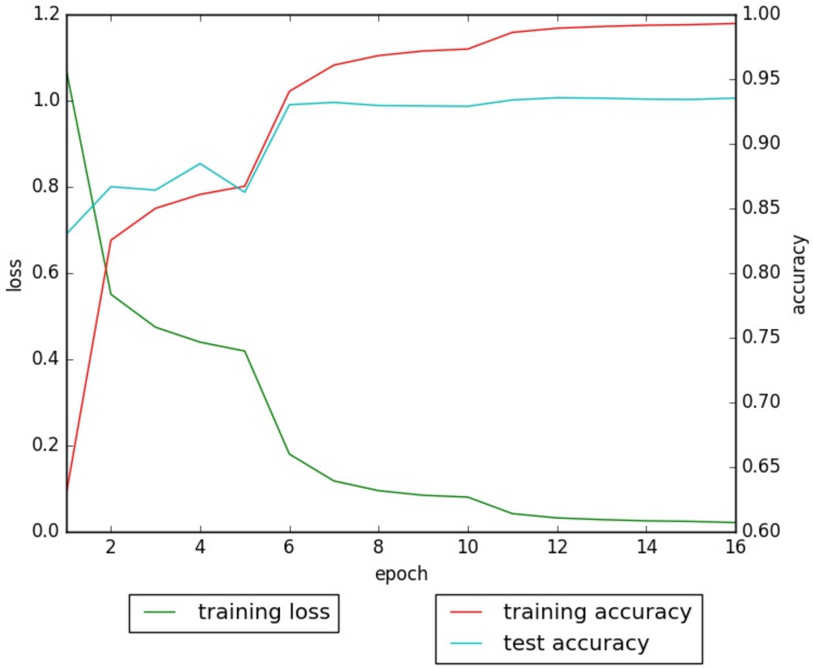

Figure 7: Training accuracy, loss and test accuracy of Visual Geometry Group neural network with 11 layers

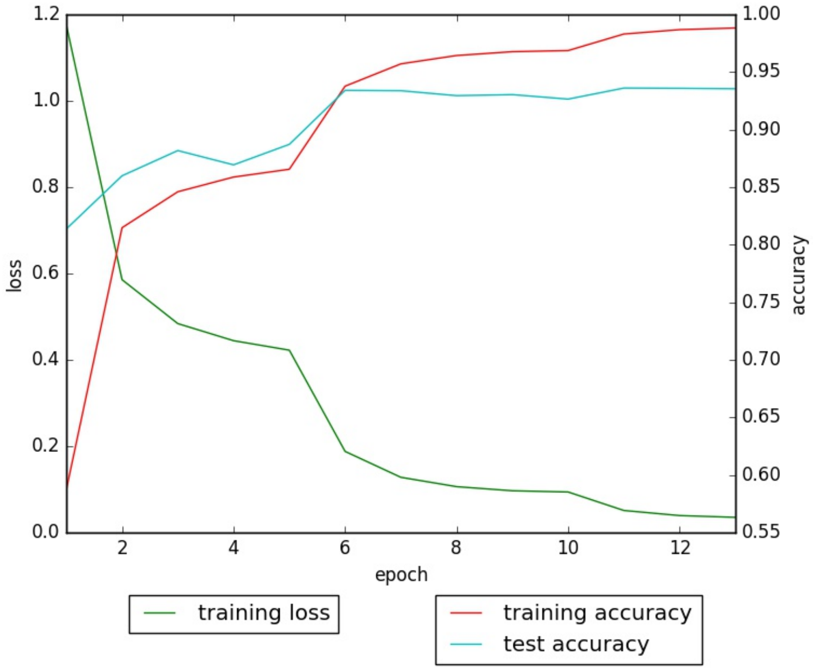

Figure 8: Training accuracy, loss and test accuracy of Visual Geometry Group neural network with 16 layers

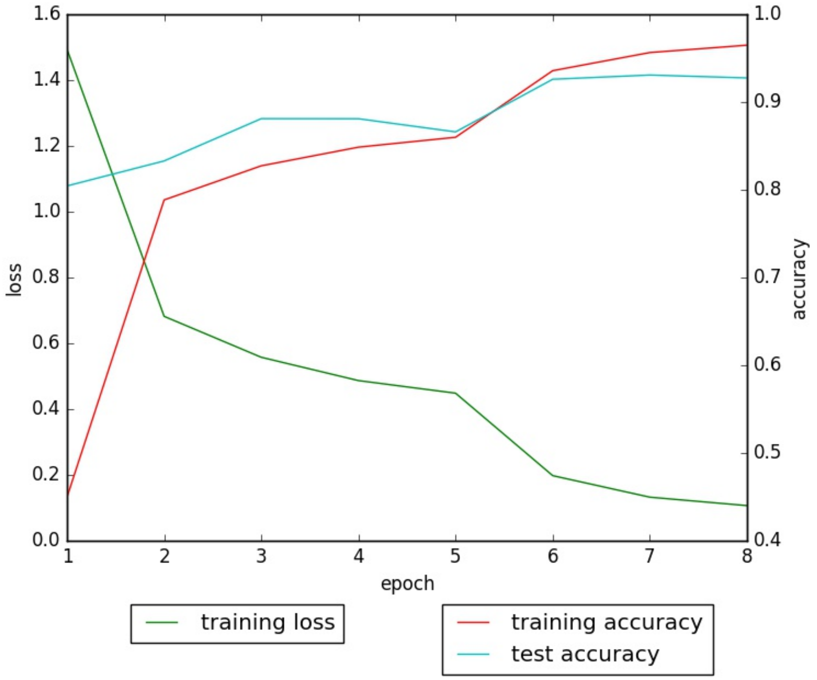

Figure 9: Training accuracy, loss and test accuracy of Visual Geometry Group neural network with 19 layers

Number of Layers	Validation Accuracy	Test Accuracy
11	0.9275	N/A
16	0.9294	N/A
19	0.9334	0.9296

Table 2 Validation Accuracy of Residual Networks and the Test Accuracy for the optimal structure among three experimenting structure at **Epoch 8**.

Table 2 reflects that as the VGG network gets deeper, it achieves higher accuracy. Thus, VGG-19 is the optimal number of layers chosen from the training process.

Discussion

From our experiment result, we can expect that for both VGG and residual network, the larger the layer number, the smaller the classification error will be in general. Thus we evaluate and compare the classification performance of 164 layer residual network and that of VGG-19 on test set.

The testing accuracy of 164-layer ResNet structure is lower than that of VGG-19 structure returned, and thus we expect that deep VGG is better than deep ResNet on the CIFAR10 classification task in general. This result does not match the result proposed in *Deep Residual Learning for Image Recognition* by He, et al. This discrepancy may due to the following two reasons: 1. In the study of He, et al, their training of ResNet utilize decaying learning rate, which incredibly increase the learning accuracy of their ResNet structure when they convert learning rate from 0.1 to 0.01. In contrast, our experiment use constant learning rate for the better comparison with VGG; 2, The training accuracy curve of ResNet reflects that the last model returned not necessarily have the highest accuracy throughout the whole training process, while in the learning curve of VGG, the last model returned does have the highest accuracy throughout the training.

Conclusion

From this experiment, we can conclude that deep convolutional neural networks are still powerful on classifying large-scaled dataset with small number of classes. Considering specific kind of convolutional neural network architectures, we expect that deep VGG networks to perform better than deep residual neural networks.

Code

https://github.com/XavierXinweiWang/COGS118FinalProject

The reference and instructions is specified in README.

Reference

Cires¸an, Dan, Meier, Ueli and Schmidhuber, Ju¨rgen. Multi-column deep neural networks for image classification. Galleria 2, 6928 Manno-Lugano, Switzerland, 2012.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing and Sun, Jian. Deep Residual Learning for Image Recognition. The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, 2016

Krizhevsky, A., Sutskever, I., and Hinton, G. E. ImageNet classification with deep convolutional neural net- works. In NIPS, pp. 1106–1114, 2012.

Simonyan, Karen and Zisserman, Andrew. Very Deep Convolutional Networks for Large-scale Image Recognition. CoRR, abs/1409.1556, 2015.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I.&Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. J. Machine Learning Res. 15, 1929–1958 (2014).