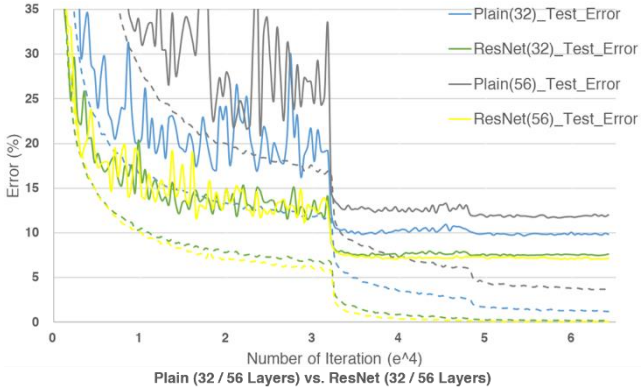
Preliminary Discussion & Experiment Overview

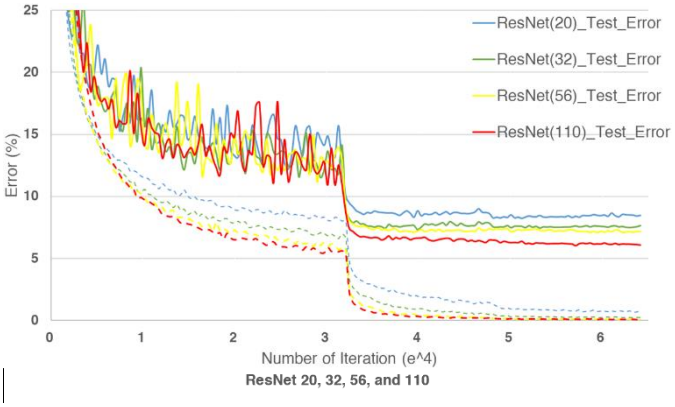
Making a deep-layered network is not only hard to train but also challenging to maintain performance. In this paper, the authors present a solution to make an even deeper layered network without jeopardizing efficiency and performance. Their solution was ***a residual block***, where the layer optimizes by ***learning the residual of output function instead of the function itself*** and enables catching even a small fluctuation of the inputs.

For the implementation, I used **CIFAR-10** dataset and followed all the settings identically (SGD optimizer with 0.1 learning rate decay by 0.1 after 32k and 48k iterations, 0.9 momentum, 0.0001 weight decay, Cross-Entropy loss). Further, my code follows a simple 2+ 6*n structure (identity shortcuts in all cases), proposed by the authors of the paper. All the models were run *five times*, and their test accuracy points (100 ï error) are reported in ñBest (MeanÑSTD)òformat. Overview of reproduced results is summarized in the following table. Moreover, logs are collected to Tensorboard. To produce a neater graph, logs are downloaded in .csv format and graphically produced as below.

Model	# Layers	# Parameters	My Reproduced Test Error (%)	Paper Reported Test Error
Plain	32 (n = 5)	464,154	9.67 (9.85Ñ0.1477)	≒9.9
Deep Plain	56 (n = 9)	853,018	11.69 (12.42Ñ0.1123)	≒13.0
Shallow ResNet	20 (n = 3)	269,722	8.22 (8.47Ñ0.2234)	8.75
ResNet	32 (n = 5)	464,154	7.22 (7.39Ñ0.1994)	7.51
Deep ResNet	56 (n = 9)	853,018	7.08 (7.19Ñ0.1311)	6.97
Deeper ResNet	110 (n = 18)	1,727,962	6.04 (6.76Ñ0.3011)	6.43 (6.61Ñ0.16)



Plain (32 / 56 Layers) vs. ResNet (32 / 56 Layers)



ResNet 20, 32, 56, and 110

Most reproduced results were similar to the paper reported results. Further, the first learning rate decay at 32k was promising, while second decay at 48k was negligible. The only difference was that reproduced results showed higher STD, especially for the Deeper ResNet (Still, most runs were better than other ResNets).

First Experiment - Major Concept of the Paper (Figure 6 Left graph from the paper)

In this experiment, I **compared Plain network to Residual network**. Two Plain and ResNet were presented here, 32 layers and 56 layers. As the authors presented, ResNet solved degradation problems compared to the Plain; while ResNet improved the performance as the layer increases from 32 to 56, Plain performed poorly. As shown in the graph, only Plain nets suffered from degradation, where both training and test errors increase as the layer depth gets deeper; thus, it is not the issue of overfitting. Further, unlike Plain, ResNets showed plateau status for both training and test errors in a similar iteration.

Second Experiment ï Going Deeper Layers (Figure 6 Right graph and Table 6 from the paper)

In this experiment, I further **explored the effects of residual connection in the deeper layers**. As presented in the paper, I was able to demonstrate the improving performance for deeper layers. For some models, reproduced errors are much better than the errors reported in the paper. However, it seemed that my reproduced results are more fluctuating by each run, as shown in the higher standard deviation. Since both training errors and test errors are showing better as the layer goes deeper, I believe the reproduction of the results are robust enough.

Limitations and Discussion

My implementation has several limitations. First, I had to choose CIFAR-10, not ImageNet, because of the training time and available hardware. I used a single GPU, while the authors used at least two GPUs. Further, while the authors suggest using different residual blocks for layers over ~100 due to the bottleneck issue on ImageNet, I did not implement such block because it was not mentioned for CIFAR-10. For ResNet(110), the authors mentioned 0.01 learning rate for at the beginning to warm up, yet, I just used 0.1, and it reached the same performance nonetheless (authors also claim that eventual accuracy was the same for both learning rate cases). I could not run ResNet(1202) due to insufficient memory on my GPU. It would have been nicer if I could delve into very deeper layers to see the effects of the residual connection. I might put that on the future work.