Paper Implementation Report

Group Normalization

Ⅰ. Implementation of Group Normalization

The key concept of group normalization is that **it allows stable learning even in small batch sizes**. In many computer vision problems, small batch sizes often require due to limited GPU resources for training models with large-scale or high-resolution images. So I tried to demonstrate the effect and characteristics of group norm compared to other normalization techniques. Though the authors used ImageNet as a benchmark, I used CIFAR10 because of the limitations of GPU resources. In the process of implementing, it was impossible to train the models through CIFAR10 because ResNet50 or ResNet101 used in the paper because it does not match with parameters and image size. To solve this issue, I implemented **ResNet56** and **ResNet110** models adapted to CIFAR10, which is mentioned at [17] in the paper. Also, as described in the paper, I initialized *Conv* layer of all models with *He Initialization* and set weight of last block's norm of ResNet to 0. In addition, I applied a scaling rule that modifies the learning rate to (0.1*N/32) according to the batch size N.

Ⅱ. Experiment results

ⅰ. Comparison of error for each normalization methods

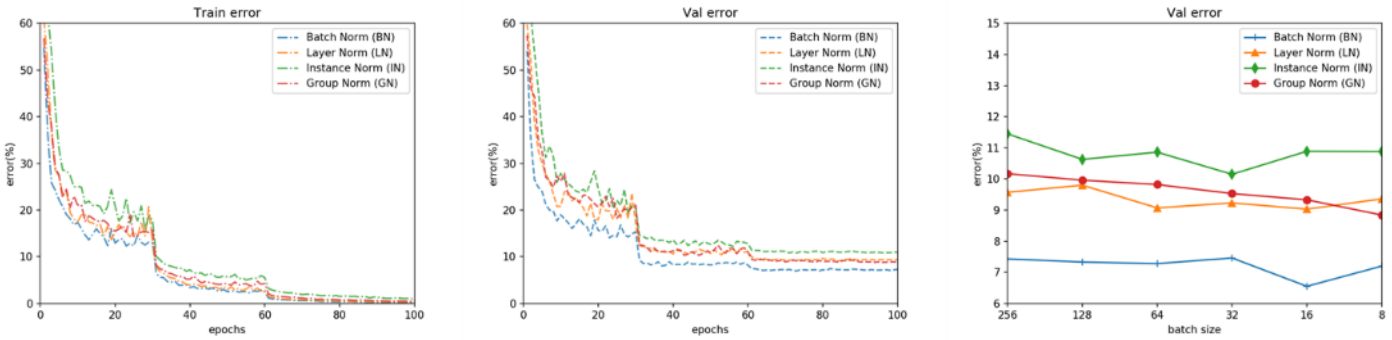
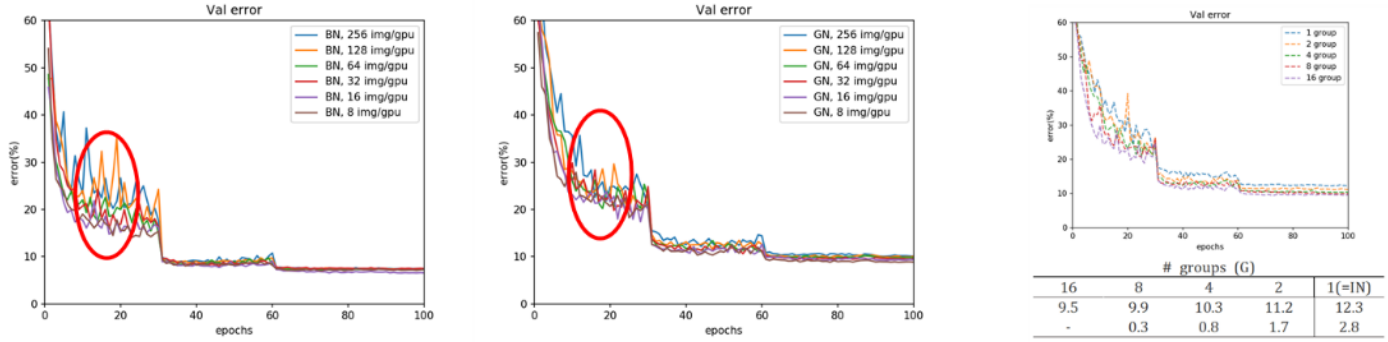


Figure 1. Comparison of error curves with a batch size of 8 images/GPU (LEFT), CIFAR-10 classification error vs. batch sizes (RIGHT)

First, we compared train and validation error for each norm when batch size was 32. Unlike expected results, in Figure 1, BN was 2% ahead of GN. **It seems that it wasn't that difficult for BN to converge because the number of parameters of ResNet56 is much smaller than that of ResNet50 for CIFAR-10, not ImageNet.** However, as a result of comparing the error by batch size, as the batch size gets smaller, the error of GN decreases gradually while the error of other Norm increases somehow in Figure 1. In addition, as you can see in Figure 2, **GN is more stable than BN even in the difference of batch size.**

ⅱ. Comparison of GN error according to group size



groups (G)

16	8	4	2	1(=IN)
9.5	9.9	10.3	11.2	12.3
-	0.3	0.8	1.7	2.8

Figure 2. Sensitivity to batch sizes (LEFT), Error curve and the median error rate of the final 5 epochs according to number of groups (RIGHT)

To find out the characteristics of Group Norm, we compared the errors by the number of groups when the batch size is 32. As shown in Figure 2, **the best performance has been achieved when the number of groups is 16. As the number of groups became smaller, the performance worsened.** For reference, in this experiment, ResNet56 had a *Conv* layer with 16 channels, so the maximum number of groups should be up to 16.

Ⅲ. Discussion

This experiment gave us some insight into the performance and characteristics of GN. However, it could not be the expected results. It seems that this may be caused by the difference of the model complexity or dataset. Also, as the authors say in the paper, we may need to experimentally find the optimal number of groups appropriate for the model and dataset. In the case of the ImageNet, the larger the # of groups, the better the performance, however, the performance decreased more than a certain # of groups. So it seems necessary to discuss **how to find the optimal number of groups.**

[17] S. Gross and M. Wilber. Based on the implementation of Training and investigating Residual Nets.