HW 05 Paper 01
Riley Densley

ImageNet Classification with Deep Neural Networks

This paper is the explanation of the Convolutional Neural Network that this team used to make a highly accurate object recognition neural network. They talk about their approach to the problem, their trials, and how they overcame some issues like overfitting.

They trained their network on the ImageNet dataset that consisted of over 15 million labeled images. This was the largest data set that had recently become publicly available. The first issue with this dataset is the mere size of it and the fact that it had over 22,000 categorize. They used a convolutional neural network because it has a large learning capacity. To further test this, more than one GPU needed to be utilized since the size of the training exercises were too large for one GPU. It is noted that having two GPU's only gave them a slight time decrease. I found this interesting. I thought that having double the hardware would significantly help with time. I suppose that the time it takes to pass the data back and forth adds to the time. I would be curious to see if a different setup in the architecture and how the data is sent back and forth would help decrease the overall time.

An issue discovered early on was overfitting. The setup had millions of parameters and each training imposed 10 bits of constraint on the mapping. This contributed to considerable overfitting. To combat this, the images were augmented by translation and reflection, then a slightly smaller portion of this image is then used to train the network further. Another method used to alter the images was to change the intensity of the image. The most effective method they found to combat overfitting was dropout. As discussed in class and in this paper, dropout sets hidden neurons to zero at a .5 probability. To me this seems like you are almost starting over with your network, but i suppose that you would still have some characteristics of your old network. I would be curious to see how the accuracy changes when a dropout is applied and how quickly the network recovers.

I found the results interesting. I am still learning about neural networks but I thought that published results would be better. A 37% error rate is fairly low. I know I need to consider the volume that was used and the thousands of categories that were needed to be classified correctly. What I was supprised the most about was the first results had

a 60% success rate. I would be interested to see results of current systems that have been tested on this same dataset.