Objective

The overarching motivation of our project is to explore techniques for achieving the highest validation accuracy. As part of this process we aim to investigate convolutional neural network (CNN) compression in hopes of increasing classification accuracy by emphasizing training on the most important filters. Hence CNN compression is another goal. Classification using CNNs has a growing influence in our lives. For example, phones can utilize CNNs for facial recognition. However, running CNNs with millions of parameters poses a problem during deployment to devices with limited computation and battery. Consequently, we also propose to optimize networks in resource limited environments.

Related Work

Deep learning models often improve with more training data. Consequently, we will use image augmentation, a common technique, for boosting the number of samples we have. In addition, [1] show that self-supervision can increase model robustness. [2] compress CNNs using partial least squares on greatly simplified feature maps (outputs of convolutional layers) to remove low-impact filters. While previous works demonstrate model compression through projections to low-dimensional subspaces, they do not utilize spatial features. Hence we propose to test the efficacy of using spatial relations through higher-order PLS to optimize inference computation of CNNs.

Technical Approach

We will increase the amount of training data by augmenting training images, such as by cropping and rotating images. We aim achieve better model robustness by developing a two headed network. One head will classify images and the other head will perform self-supervised learning - randomly cropping portions of the image out and predicting the missing sections.

We plan two novel methods for network compression. One is utilizing higher order partial least squares on convolutional layer outputs and the class labels to project the outputs into a lower dimensional latent space. We then identify filters with the lowest importance using some variable selection method, such as variable importance in projection [3].

A second method is to identify the filters whose output activations contribute the least to the output of the network before the softmax layer. This can be done by finding the absolute value of derivative of the final output of the network with respect to the activations of each filter through backpropagation, and taking the average contribution of each filter's output activation over the entire training set. If such a derivative is low, it means that the output of a particular filter is not used much in later parts of the network, so the filter contributes little to the overall result. We take the average so that we find the filters that generally contribute little - not just filters that are activated only in certain classes, but filters that are rarely activated.

We then prune the low-impact filters by only copying the weights of the remaining filters to a new CNN architecture which does not contain the low-impact filters. We then fine tune the network by training over several more epochs. We repeat this pruning and fine tuning process either until the network loses validation accuracy or we remove a preset

percentage of the filters.

[1] Dan Hendrycks et al. “Using Self-Supervised Learning Can Improve Model Robustness and Uncertainty”. In: *Advances in Neural Information Processing Systems (NeurIPS)* (2019).

[2] Artur Jordao, Fernando Yamada, and William Robson Schwartz. “Deep network compression based on partial least squares”. In: *Neurocomputing* 406 (2020), pp. 234–243. ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2020.03.108. URL: https://www.sciencedirect.com/science/article/pii/S0925231220305762.

[3] Tahir Mehmood et al. “A review of variable selection methods in Partial Least Squares Regression”. In: *Chemometrics and Intelligent Laboratory Systems* 118 (2012), pp. 62–69. ISSN: 0169-7439. DOI: https://doi.org/10.1016/j.chemolab.2012.07.010. URL: https://www.sciencedirect.com/science/article/pii/S0169743912001542.