CS360 Machine Learning Final Competition

Kangrui Yu

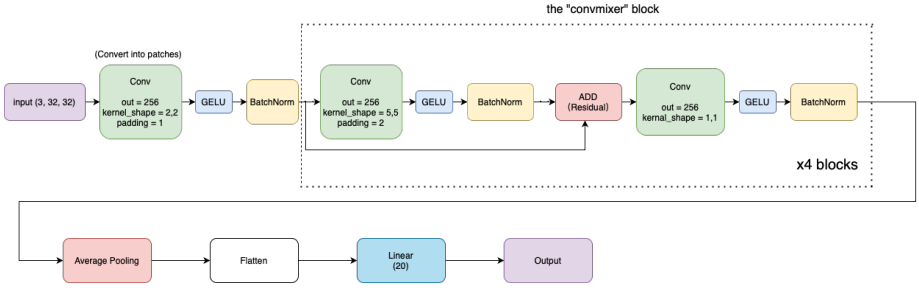
December 2023

1 Introduction

1.1

The model I use originates from a paper called “Patches are all you need” by Asher Trockman, J. Zico Kolter. The intuition for using this structure is that it doesn't contain pooling layers except for the first convolution used to convert image into patches, which I think is an advantage for small image of 32x32. In addition, the patching procedure cuts the image into several smaller parts, which I think aligns with the human instinct of image classification. I utilized the ”convmixer” block introduced in the paper.

The network takes patches as inputs(which are 256 2x2 segments of the original image) and uses depth-wise convolution(one filter for one channel) and point-wise convolution(1x1xchannels filter), and a residual block. Here's a flow chart of the Network(Total of 10 Computational Layer, 302,868 parameters):



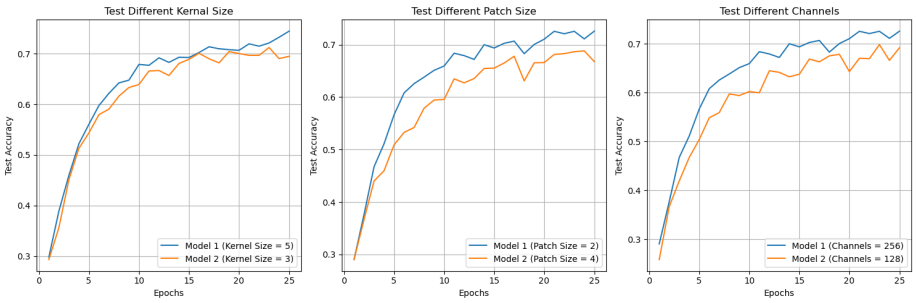
1.2

GPU : Apple M2 Pro
GPU Time: Approximately 34.1 seconds per batch(batch size = 512)
I trained 150 epoches, which is a total of about 85.25 mins
The final test score : 0.8348 (23th on the leaderboard)

2 Training Procedure

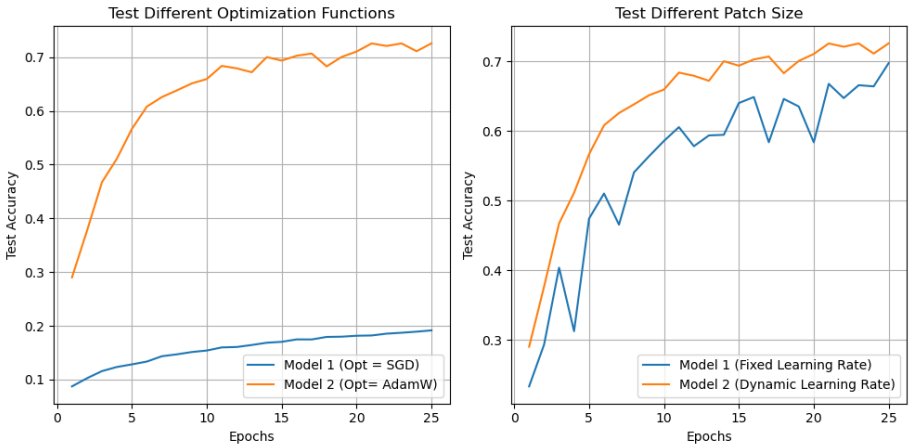
The hyperparamters I tuned in my model contains kernal size, number of channels, optimization function, learning rate and the use of different data augmentation methods.

2.1 kernal size, patch size, number of channels



Kernal size = 5x5, patch size = 2, number of channels = 256 tested to have better performance on the test set and converges faster.

2.2 Optimization function, Learning rate



AdamW Optimization Function with a dynamic learning rate converges faster.

2.3 Data Augmentation

In my model, I used RandomResizedCrop, RandomHorizontalFlip, RandAugment, ColorJitter, and RandomErasing.I found that using all the data augmentation methods outperform using some of them.

3 Works Cited

arXiv:2201.09792 [cs.CV]