

# Experiments with Convolutional Neural Network in TensorFlow

## Background

Convolutional neural network is a powerful learning mechanism that was applied successfully to many computer vision tasks. In this project we will run experiments using `tensorflow`. The experiments are closely related to the online example available at:

[https://www.tensorflow.org/get\\_started/mnist/pros](https://www.tensorflow.org/get_started/mnist/pros).

The above online example shows how to apply a convolutional neural net to learn how to classify data from the MNIST dataset. The code in the example takes about 30 minutes to train, and achieves over 99% accuracy.

In this project you are asked to solve the exact same problem: creating a classifier for the MNIST data. However, we put two constraints on the network being created that are intended to make **the problem more challenging** and **shorten** the training time:

1. The first layer in the network must be a  $4 \times 4$  **maxpooling** layer.
2. There is a limit on the number of batches and the batch size. If  $n_1$  is the number of batches and  $n_2$  is the batch size, your program must satisfy  $n_1 \cdot n_2 \leq 100,000$ . For example, you can have  $n_1 = 2000$ , and  $n_2 = 50$ .

## Example program

An example program is available as `proj3.py`.

## Grading

Your grade will be based on the accuracy of your model on the testing data. Observe that because of the random initialization different runs of the program may produce (slightly) different accuracy values. The one that counts for your grade is the one produced by your program when you meet with the TAs.

## What you need to submit

1. Source code of the python script.
2. Documentation describing your experiments and the test accuracy that your program achieves.