INC491 Homework 3: Image classification using Convolutional Networks (CNN)

Due: Oct 30, 2022 (in LEB2)

What to submit: (in a single pdf file)

1. Diagram of your network and parameter used (such as size, no. of neurons, shape of data in each layer etc.)
2. Printed Source code
3. Graph Results (4 graphs)

In this homework, you have to train a convolutional neural network to classify images from the CIFAR10 dataset. Information about the CIFAR10 dataset is in the URL below.

 https://www.cs.toronto.edu/~kriz/cifar.html

CIFAR10 dataset consists of images of 10 classes, which are airplanes, automobiles, birds, cats, deers, dogs, frogs, horses, ships, and trucks. Each image is a color RGB which has a size of 32x32. We can use Keras library to download the dataset by commands below.

```
cifar = tf.keras.datasets.cifar10

(xtrain, ytrain), (xtest, ytest) = cifar.load_data()
```

The data is divided into 50,000 images training and 10,000 images for testing. Please examine the data after you download it. For example, use the command below to show the 1st picture of the training data.

```
import matplotlib.pyplot as plt

plt.imshow(xtrain[0])

plt.show()
```

Design your neural network to learn these images. Roughly draw a diagram that shows the number of layers, their parameters, and the shape of data in each layer (submitted item 1). Train and print the graph of loss and accuracy on the training set and the test set along the training process (total of 4 graphs from tensorboard). Please note that your network and parameters should achieve a reasonable accuracy in prediction. You have to try and find your own parameters. The training process will be quite long to achieve a good performance.

1. loss of the training set

2. loss of the test set

3. Accuracy of the training set

4. Accuracy of the test set

Additional suggestions.

- You should normalize the data before training.
- Use an appropriate batch size.
- Find information about parameters in function "fit". It uses default parameters. See if adjusting any of these can help you.
- Prevent overfitting as necessary.
- Use Google Colab if your computer is slow or no GPU. I do not expect you to train more than 4 hours. Be careful on the limit usage of Colab's GPU. Use small number of epochs and do not use GPU yet when you debug your code.