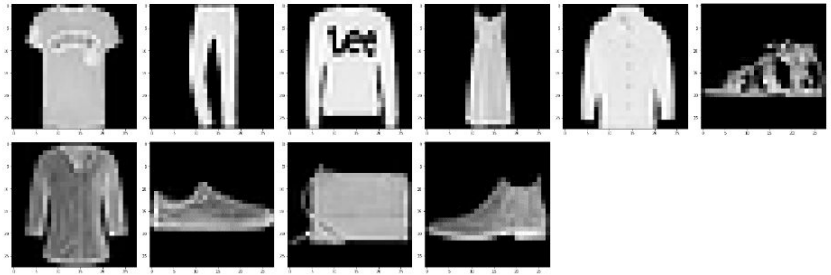
COEN 140L Lab 8 Intro

Xuyang Wu

Dataset



```
import tensorflow as tf
from tensorflow import keras

# import the data
mnist = keras.datasets.fashion_mnist

# load the data, train = (60000,28,28), test = (10000,28,28)
(x_train, y_train), (x_test, y_test) = fashion_mnist.load_data()

train_images = train_images.reshape(60000, 28, 28, 1))
test_images = test_images.reshape(10000, 28, 28, 1))

# normalize the pixel values to be real numbers in [0,1], It's fine if you don't normalize them
x_train, x_test = x_train / 255.0, x_test / 255.0
```

Implement convolution layer in Tensorflow

```
# initialize the model within Tensorflow package
model = models.Sequential()

# adding the convolution layer to model, layer 2
model.add(layers.Conv2D(32, (3, 3), activation='relu', padding='same', strides = (1,1), input_shape=(28,
28, 1)))

# adding the max pooling layer to model, layer 3
model.add(layers.MaxPooling2D((2, 2), padding='same', strides = (2,2)))

# … to be completed by yourself
```

Calculate the number of parameters and number of multiplications

- **9 layers in this assignment, you need give the results for each layer separately.**
- **Example: 2d-convolutional layer: filter size 3x3, depth=32, padding='same', strides = (1,1), ReLU activation function**
  - Layer-2: 2d-convolutional layer: filter size 3×3, depth=32, input size: 28×28×1
  - Param: (3×3×1+1) ×32=320
  - Output dimension of this layer: 28×28×32
  - Multiplications: (3×3×1+1) ×(28×28×32)=250,880 # for 1 neuron, 3×3×1+1 multiplications are needed, 1 is for the bias, (28×28×32) neurons as the output

Tips

- **Comparison of recognition accuracy between Lab 7 and Lab 8**
  - For training, using the **same batch_size** as that used in Lab 7, default **batch_size=32**

Include in the report

1. Give the recognition accuracy rate of the whole test set, and show the confusion matrix.
2. For each layer of the network:
   a. Manually calculate the number of parameters of that layer. That is, the number of weight elements (include the bias terms). Show you work. Verify whether your results are the same as those given by model.summary().
   b. Write out the output dimension of that layer.
   c. Manually calculate the number of multiplications required to generate that layer. The weights have bias terms. You only need to consider the multiplications required to calculate the weighted sums. You don't need to consider the multiplications involved in the softmax function. Show you work.
3. Compare the recognition accuracy rate of the test set, total number of parameters, and total number of multiplications of this CNN to those in Lab 7 (neural network). Analyze your findings and explain why you obtain different (or similar) results.

Lab Tasks

- **Need demo** for week 8 assignment(10% points).
- Submit to Camino a pdf report with answers(70% points), the report contains some **results** which required by lab document, you also need to add some **observations** for the questions.
- Submit all the source code needed to generate these answers to Camino(20% points).