UNIVERSITY OF LINCOLN

CMP9137M Advanced Machine Learning

Workshop 2: Feedforward Neural Networks_BP_CNN

Access Code:

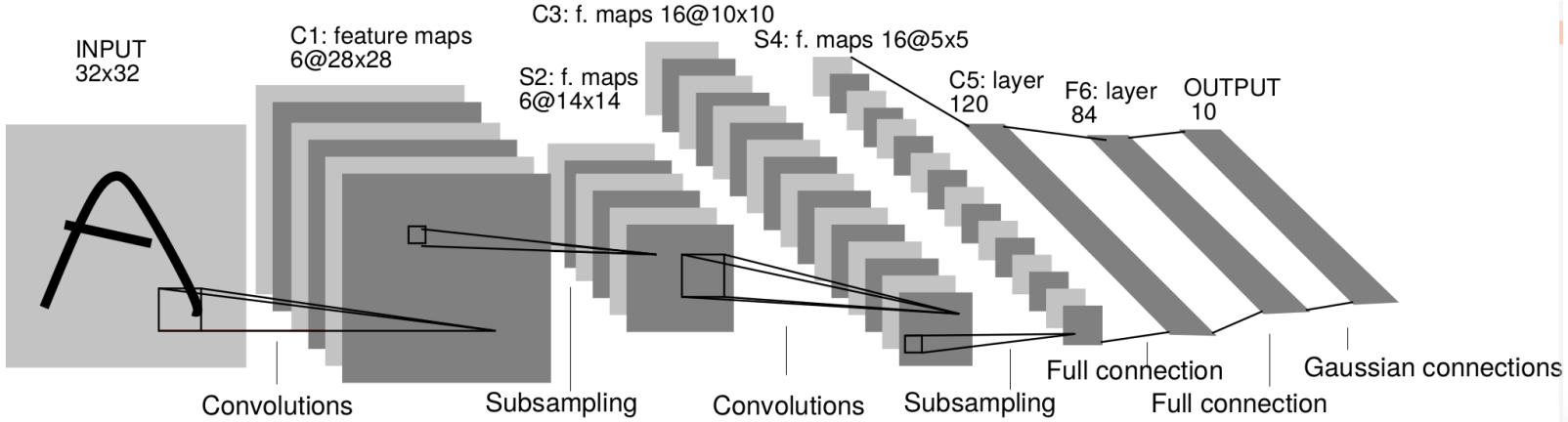
Laboratory of Vision Engineering

Dr. Lei Zhang

lzhang@Lincoln.ac.uk

Aim: workshop 2

The aim of this workshop is to gain practical experience with Convolutional Neural Networks (CNNs)-example CNN below



LeCun, Y. et al., (1998), \Gradient-Based Learning Applied to Document Recognition", IEEE

Task 1: workshop 2

Consider a Convolutional Neural Network for image classification that receives inputs of 40x40x3 pixels and outputs a corresponding label out of 10 possible outputs. The First hidden layer (CONV1) has 8 filters, filter size 5, stride 1 and pad 2. The 2nd hidden layer (POOL1) has filter size 2 and stride 2. The 3rd hidden layer (CONV2) has 16 filters, filter size 5, stride 1 and pad 2. The 4^{th} hidden layer (POOL2) has filter size 2 and stride 2. The last two hidden layers (FC1, FC2) are fully connected with 300 and 100 nodes, respectively. Note that the output layer has 10 output units.

- Draw an example architecture of this deep neural network.
- What is the total number of weights per layer and in total?

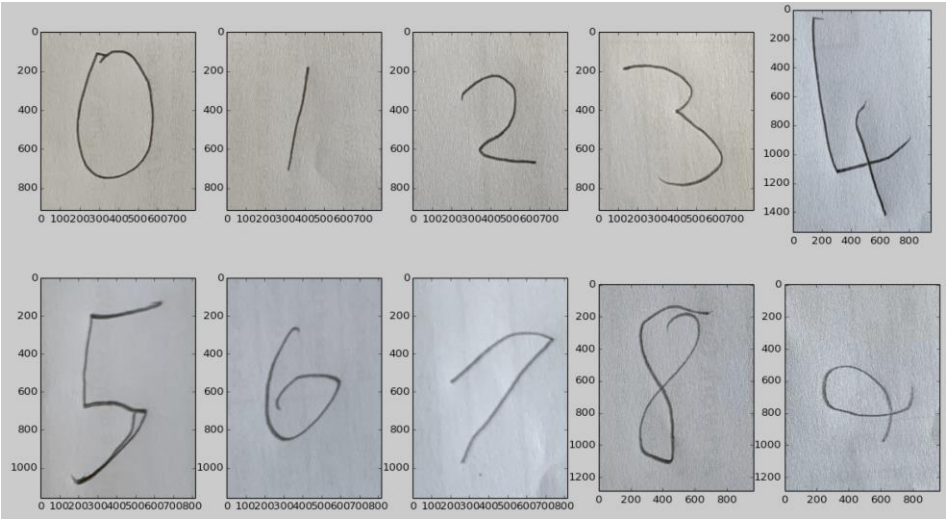
Task 2: workshop 2

Run the mnist_cnn.py program (example output below), available under the workshop materials of week 2 on Blackboard. Did you get better classification accuracy than mnist_mlp.py (from last week)?

```
 visible gpu device (device: 0, name: GeForce GT 430, pci bus id: 0000:01:00.0) with Cuda compute capability 2.1. The minimum r
equired Cuda capability is 3.0.
60000/60000 [==============================] - 128s 2ms/step - loss: 0.2656 - acc: 0.9178 - val_loss: 0.0601 - val_acc: 0.9796
Epoch 2/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0886 - acc: 0.9737 - val_loss: 0.0402 - val_acc: 0.9859
Epoch 3/12
60000/60000 [==============================] - 129s 2ms/step - loss: 0.0658 - acc: 0.9800 - val_loss: 0.0328 - val_acc: 0.9895
Epoch 4/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0540 - acc: 0.9838 - val_loss: 0.0369 - val_acc: 0.9866
Epoch 5/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0470 - acc: 0.9860 - val_loss: 0.0283 - val_acc: 0.9900
Epoch 6/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0419 - acc: 0.9875 - val_loss: 0.0299 - val_acc: 0.9906
Epoch 7/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0371 - acc: 0.9891 - val_loss: 0.0268 - val_acc: 0.9910
Epoch 8/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0348 - acc: 0.9894 - val_loss: 0.0278 - val_acc: 0.9907
Epoch 9/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0323 - acc: 0.9905 - val_loss: 0.0278 - val_acc: 0.9907
Epoch 10/12
60000/60000 [==============================] - 127s 2ms/step - loss: 0.0305 - acc: 0.9902 - val_loss: 0.0274 - val_acc: 0.9911
Epoch 11/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0288 - acc: 0.9910 - val_loss: 0.0282 - val_acc: 0.9900
Epoch 12/12
60000/60000 [==============================] - 128s 2ms/step - loss: 0.0276 - acc: 0.9913 - val_loss: 0.0283 - val_acc: 0.9910
Test loss: 0.02828781932545462
Test accuracy: 0.991
--- 1541.0500259399414 seconds ---
```

Task 3: workshop 2

Save the model[2] generated from Task 2 and use it to test[3] the data in “TomsDigits.zip" - available under the workshop materials of week 2 in Blackboard. What classification accuracy did you get?



[2]model.save weights("mnist-cnn-model.h5"), [3]model.load weights("mnist-cnn-model.h5")