King Fahd University of Petroleum and Minerals

Department of Information and Computer Science

ICS71 ANN and Deep Learning (211)

Homework 3

Date: 16/10/2021 Due date: 30/10/2021

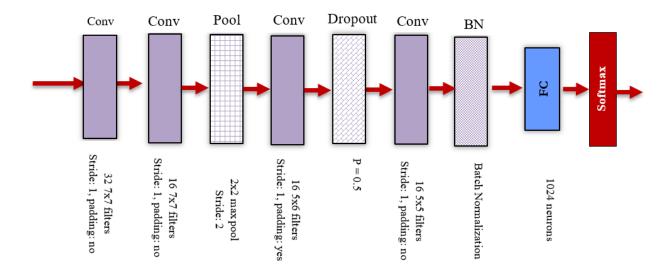
Face recognition can be categorized into face classification and face verification. Given an image of a person's face, the task of classifying the ID of the face is known as face classification, which is a closed-set problem. The task of determining whether two face images are of the same person is known as face verification, which is an open-set problem¹.

In this assignment, you will use Convolutional Neural Networks (CNNs) to design an end-to-end system for face classification/identification. Your system will be given an image as input and will output the ID/name of the person shown in that image.

You will train your model on a dataset with a few thousand images of labeled ID's (i.e., a set of images, each labeled by an ID that uniquely identifies the person). Use Jupyter notebook to show each of the following steps:

<u>Part 01 [80 points]</u>

- 1. Prepare the data by:
 - a. Preprocessing the data by zero-centering it
 - b. Dividing the train data into train (80%) and validation (20%)
- 2. Implement the following model in Pytorch. The specifications of each layer are given under it.



¹ For close-set task, all testing identities are predefined in training set. For open-set task, testing identities typically do not appear in training set

- 3. Babysitting the experiment as discussed in the class and show the output of each step.
- 4. Perform a random search to select the following model's parameters: Learning rate, Regularization strength.
- 5. Show the performance of the following optimizers: Gradient Descent, SGD, AdaGrad, RMSProp, and Adam. Then, use the one with the best performance for the following steps.
- 6. Train the model on the whole data and report the results using the test data
 - a. Use early stopping to avoid overfitting
 - b. Show the loss and accuracy of the train and validation data
 - c. Plot the loss of train and validation data in one figure
 - d. Plot the accuracy of train and validation data in one figure
 - e. Show the accuracy of the test data
- 7. What changes to the network architecture can improve the results? Show the modified model with the results.

Part 02: Transfer Learning [20 points]

In this part, you will use transfer learning to extract features from the dataset using VGG-16 and feed them into **FC** and **Softmax** layers (last layers of model shown in Part-01) for classification. The new model will have all the layers of VGG-16 up to the first FC layer with 4096 neurons. The output of VGG-16 will be used as an input to the FC layer followed by Softmax classifier. You need to train only the last layers of your model on our dataset. Submit the code with results in the same Jupyter notebook file.