Project 1   **Due 01/25/2021**

1 Introduction

This is a group project. Each group should consist of 2 or 3 students. The aim of this project is to implement different classifiers to achieve face recognition. You are given a set of faces and their corresponding labels. You need to split the data into training and testing sets and use the training data to train your classifiers. The whole pipeline is described in the following section.

2 Classifiers

Preprocessing

1. **PCA.** You have to implement Principal Component Analysis (PCA) and apply it to the data before feeding it into the classifiers above.

2. **MDA.** Similar to PCA, you have to implement MDA followed by the training/application of the classifiers.

Classifiers

1. **Bayes' Classifier.** Assuming the underlying distribution is Gaussian, you need to implement the Maximum Likelihood estimation with Gaussian assumption followed by Bayes' classification.

2. k**-NN Rule.** Implement the k-Nearest Neighbors (k-NN) rule to classify the test data. Vary k to see its effect and discuss it in the report.

You can try different scenarios. For example, you can test the effect of illumination variations, etc. Besides, you can vary the ratio between training data and testing data. Feel free to play with the data!

3 Datasets

You are provided with the following datasets (.mat files). For each dataset, you should divide it into two disjoint sets, one for training and another for testing.

> **DATA** Cropped images of 200 subjects, 3 images each, each image of size 24x21. The file data.mat has a variable `face` of size (24x21x600). The images corresponding to the person labeled n, n = 1, . . . , 200 can be indexed in Matlab as `face(:,:,3*n-2)`, `face(:,:,3*n-1)` and `face(:,:,3*n)`. The first image is a neutral face, the second image has a facial expression, and the third image has illumination variations. You can read this set using the following Matlab script:

```
data = load('data.mat');
face = data.face;
face_neutral = face(:,:,1:3:end);
face_exp = face(:,:,2:3:end);
face_illum = face(:,:,3:3:end);
```

POSE Cropped images of 68 subjects under 13 different poses. You can access the i_{th} pose of the j_{th} subject as `pose(:, :, i, j)`. Here is a MAtlab script for reading it:

```
data = load('pose.mat');
pose = data.pose;
```

ILLUMINATION This is an additional dataset for POSE. Here is a script for reading it in Matlab and displaying all images for the first subject:

```
data = load('illumination.mat');
>> illum = data.illum;
% display the first subject
figure;
colormap gray
for j=1:21 subplot(3,7,j);imagesc(reshape(illum(:,j,1),[48,40])); end
```

4   Classifying tasks

Neutral faces vs smiling faces. Dataset `data.mat`. Use PCA for preprocessing and (i) Bayes's and (ii) k-NN classifiers. Experiment with (a) splitting the dataset to training and test subsets, (b) the dimension of the PCA space, and (c) with number k of nearest neighbors. In each experiment, find the numbers of images classified correctly and incorrectly.

   Identifying subjects. Dataset `pose.mat`. Use PCA and/or MDA for preprocessing and (i) Bayes's and (ii) k-NN classifiers. Feel free to start with just a few subjects. You can use the first 10 poses for each subject for training set and the last three for test set. Experiment with preprocessing and with classifiers trying to maximize correct classification. You also might want to experiment with illumination effects.

5   Programming language

Use any suitable high-level language: Matlab, Python, etc. You should not use any online toolbox that provides the classifiers above. **Every student must develop an individual set of all codes.** However, please feel free to communicate with each other and learn from each other.

6   Submission Guidelines

Please submit one report per working group of 2–3 students with descriptions of what has been done, choice of parameters, summaries of your observations, figures, tables, and conclusions. Every

group member should link her/his codes to the report pdf. These can be e.g. Dropbox links or GitHub links, etc. Please DO NOT upload your codes on ELMS, only pdf files with links to your codes.

Reports must be prepared in LATEX or any other text editor. *Handwritten reports will not be graded.*