ASSIGNMENT 2: FACE RECOGNITION

TUTOR: STELIOS ASTERIADIS
TEACHING ASSISTANT: ESAM GHALEB

DEADLINE: 28/05/2019 AT 08:30 AM

DESCRIPTION

Use the nearest neighbor recognition algorithm to perform face recognition on the dataset of the faces.zip file that you will find on *Eleum*. In particular, *ORL_32x32.mat* contains *10 32x32* different images for *40* people. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). Moreover, you will find three more folders *(3Train, 5Train, 7Train)* which account for the number of training images per person (to be used in your project) and are actually the corresponding indices for each one of the *400* images, classifying them as training or testing samples.

STEPS

Before applying the method, scale all image intensities so that they range from *0* to *1*

1. Apply eigen-analysis as discussed in the lecture (and also the book of Szeliski) and plot out the first K principal components (use the reshape function).

2. Project the training data on the new space and derive descriptors $w_{tr,i}$ for each training image *i*. Show how a trained image can be reconstructed using eigenvectors and $w_{tr,i}$. This reconstruction will help you chose the right K for your experiments.

3. Project the test data and derive descriptors $w_{te,i}$.

4. Implement a *NN* technique for face recognition using your descriptors. Discuss the results and the accuracy you obtain (fraction of test images correctly classified).

Comment on your results for a varying number of *K* (indicatively, *10, 20, 30*) and a varying number of training samples per person. Create plots for each classification rate.

DELIVERABLES

1. A report of *400-1000* words with plots and figures.

2. Well documented and commented *Python/MATLAB* code clearly separating the training and the testing part as two different functions.

3. Upload everything on *Eleum* as a zip file