Due: 1^{st} **May 2017 11:55 PM** Total points: 20

This assignment contains quiz part and programming part.
For the quiz part, the total time allowed is 60 minutes. Entire quiz part has to be taken in one session. No peeking at the questions before you begin the exam. No internet, no books, no class notes allowed during the exam Along with the questions and answers provide the following in your submission:

- Student Name
- Date you took exam
- Time exam started
- Time exam ended
- Signed Honor statement declaring that you followed all the exam rules.

The password for the quiz part is HONORCODE.
For the programming part, the guidelines are at the end of the assignment. You are going to design a face identification system using:

1. Standard Principal Component Analysis (PCA)
2. Linear Discriminant Analysis (LDA)

Face Database

In this assignment, you are given the ORL face database, containing N = 40 subjects with 10 face images per subject. Let the subjects be denoted as $\mathcal{D}_s$, s = 1,2,.,N. The database is provided as a Matlab `.mat` file, `faceDat.mat`. It contains a Matlab struct, `faceDat`. For instance, the third face image of subject 27 ($\mathcal{D}_{27}$) can be accessed as: `I = faceDat(27,3).Image`. Since images are stored as unsigned integers (`uint8`), you might find it useful to convert them to `double` format by: `I = double(I)/255`; or succintly `I = im2double(I)`; (This code also maps the image pixel values from [0 255] to [0 1]).

Training and Testing Set

You will use the first five images of of each subject as the *training set*. The remaining five images of every subject will be used as a *test set*. You are going to find the identity of images in the test set.

1 Using PCA

Steps of feature extraction method are as follows:

1. Given a set of M face images: $I_1, I_2, ...I_M$, vectorize the pixel values to obtain pixel column vectors X_i. If the row and column size of an image I_i is $P \times Q$, then the size of X_i is $(P \times Q) \times 1$. Let R be $R = P \times Q$, for simplicity. You can vectorize a face image by simply concatenating each pixel row side by side. X_i will also be referred to as a face image from now on.

2. Form the *scatter matrix* S as

$$S = \sum_{i=1}^{M} (X_i - \mu)(X_i - \mu)^T$$

$$\mu = \frac{1}{M} \sum_{i=1}^{M} X_i$$

   The size of S is $R \times R$

3. Compute the eigenvectors, u_k, and eigenvalues λ_k of S. The size of an eigenvector u_k is $R \times 1$. If you de-vectorize u_k, to obtain $P \times Q$ image (the reverse operation you perform in Step.1), you get an *Eigenface*.

4. Select K eigenvectors, $u_1, u_2, u_3, ...u_K$, having the biggest corresponding eigenvalues: $\lambda_1 > \lambda_2 > \lambda_3 > ... > \lambda_K$ and form the PCA transformation matrix $U = [u_1, u_2, ...u_K]$. The size of U is $R \times K$.

5. *PCA Projection*: Project (or extract the feature of) a given face image, X_i, into the PCA space by: $Y_i = U^T X_i$. The size of Y_i is $K \times 1$.

Above, steps from 2 to 4 is called *PCA training*. The output of PCA training is the PCA transformation matrix U. Step 5 explains how to use U to extract PCA features (principal component coefficient vectors) from a given face image.

1.1 Training Phase

Use all the images in the training set to compute U. Using U, compute the PCA features for all the images in the training set. Let these features be $Y^{tr}_{s,1}$, $Y^{tr}_{s,2}$, $Y^{tr}_{s,3}$, $Y^{tr}_{s,4}$, $Y^{tr}_{s,5}$ for subject $\mathcal{D}_s$.

1.2 Testing Phase

Using U, compute the PCA features of images in the test set. Let Y_t be the feature vector of a test image. Assign the test image to that subject $\mathcal{D}_s$ such that $\frac{1}{5}\sum_{j=1}^{5} ||Y_t - Y^{tr}_{s,j}||$ is minimum. At the end of this phase, each test image has been classified as a subject according to the above metric. You also know the ground truth which is given by the database. Hence, Identification Percentage is defined as:

$$\begin{aligned}\text{Identification Percentage} &= \frac{\text{Number of test images correctly identified or classified}}{\text{Total number of test images}} \\ &= \frac{\text{Number of test images correctly identified or classified}}{200}\end{aligned}$$

$||.||$ denotes Euclidean distance.

1.3 Tasks 1

1. Display the first 20 Eigenfaces **1 point**
2. Plot the Identification Percentage vs PCA dimensionality (K) for $K = 5$, 10, 15, 20, 25,... 2500. **2 point**
3. Provide a table having columns as *PCA feature vector size (K)*, *Total variance explained*, *Identification Percentage*. Provide the table for $K = 5$, 10, 20, 40, 60, 100, 150, 200, 400, 1000, 2000. The total variance explained is computed by

$$\text{Total variance explained} = \frac{\sum_{i=1}^{K} \lambda_i}{\sum_{i=1}^{R} \lambda_i}$$

2 point

2 Using LDA

Steps of feature extraction method are as follows:

1. Given a face image I, vectorize the pixel values to obtain pixel column vectors X.
2. Within-class scatter matrix S_W:

$$S_s = \sum_{X \in \mathcal{D}_s} (X - \mu_s)(X - \mu_s)^T \text{ where } \mu_s = \frac{1}{\#\mathcal{D}_s} \sum_{X \in \mathcal{D}_s} X$$

$$S_W = \sum_{s=1}^{N} S_s$$

3. Between-class scatter matrix

$$S_B = \sum_{s=1}^{N} (\#\mathcal{D}_s)(\mu_s - \mu)(\mu_s - \mu)^T \text{ where } \mu = \frac{1}{\sum_{s=1}^{N} \#\mathcal{D}_s} \sum_{allX} X$$

4. Compute the eigenvectors, u_k, and eigenvalues λ_k of $S_W^{-1}S_B$.

5. Select K eigenvectors, $u_1, u_2, u_3, ...u_K$, having the biggest corresponding eigenvalues: $\lambda_1 > \lambda_2 > \lambda_3 > ... > \lambda_K$ and form the matrix $U = [u_1, u_2, ...u_K]$.

6. *LDA Projection*: Project (or extract the feature of) a given face image, X, into the LDA space by: $Y = U^T X$.

Using the new U, follow the same procedure for training and testing as given previously.

2.1 Tasks 2

1. S_W turns out to be a singular matrix and non-invertible. This results in complex eigenvalues and eigenvectors for $S_W^{-1}S_B$. Give an explanation to why S_W is singular for the training set. **1 point**

   Use Moore-Penrose pseudo-inverse of S_W, (MATLAB command: `pinv(S_W)`), for rest of Task 2.

2. Display the first 20 Eigenfaces **1 point**

3. Plot the Identification Percentage vs LDA dimensionality (K) for K = 5, 10, 15, 20, 25,... 1000. **2 point**

4. Provide a table having columns as *LDA feature vector size (K)*, *Total variance explained*, *Identification Percentage*. Provide the table for K = 5, 10, 20, 40, 60, 100, 150, 200, 400, 1000, 2000. **2 point**

3 Task 3

1. Compare and explain the performances of PCA and LDA. **1 point**

2. Give a solution to solve the problem mentioned in Task 2. **1 point**

3. (Research on-line or Propose) and explain a method combining PCA and LDA to create a better Face Recognition system **2 point**

Tips

- For performance plots (accuracy vs PCA/LDA dimensionality) required, you do not need to re-compute eigenvectors/eigenvalues at each dimensionality. Compute them once and use the *required part* of the eigenvector matrix for feature extraction. This saves a lot of time.

- When plotting images/eigenvectors, use `imshow()` function. If the displayed image is not meaningful, try `imshow(image,[])`.

Certain matrix operations in the assignment are computationally intensive, so it is recommended to start this assignment early.

4 *Grad Credits*: Head Pose Estimation

5 point

Read the attached paper on head pose estimation and briefly describe 5 methods of your choice.

Submission Instructions

Every student must submit following 2 files:

- An organized report submitted as a PDF document. The report should describe the implementation, issues (problems encountered, surprises), and an analysis of the test results (interpretation of effects of varying parameters, different image results). Intermediate and final results must be provided.
- A ZIP file containing the necessary codes.

The heading of the PDF file should contain the assignment number and topic. Also, attach a photo of yourself at top-left of the PDF along with your name and department.

Late Submission Policy

No late days are allowed for this assignment.

Collaboration Policy

I encourage collaboration both inside and outside class. You may talk to other students for general ideas and concepts but the programming must be done independently.
For the written part, there will be no collaboration permitted.

Plagiarism

Plagiarism of any form will not be tolerated. You are expected to credit all sources explicitly. If you have any doubts regarding what is and is not plagiarism, talk to me.