

**ECE 442 Introduction to Multimedia Signal Processing Winter 2021**  
**Laboratory #2, PCA and Eigenfaces, February 12, 2021**  
**Report Submission Due Date: March 2, 2021**

---

1. Please submit in softcopy at eclass by 5 pm of March 2<sup>nd</sup> as a single zip file in FirstName\_LastName\_lab2.zip. The zip file is expected to contain the following files: your answer sheet, Matlab code, and input/output files to be reproducible on our side.
2. Questions that should be answered in your lab report are numbered and marked as bold within the following text. Please number your answers accordingly. Cite any material that you have used (either it is a website or a paper or a piece of code).
3. Certain questions ask for images to be uploaded. To save images from MATLAB to the file system, use `imwrite(Im, 'filename.bmp')`. Do this instead of saving an image from a figure window. Always use a lossless extension, e.g. bmp or png.
4. Make sure your Matlab code is bug-free and works out of the box. Please be sure to submit all main and helper functions. Be sure to not include absolute paths. Points will be deducted if your code does not work at our side.

## **Principle Component Analysis and Eigen Faces**

### **1. PCA**

- 1.1. Suppose you are given a set of  $(x_i, y_i)$  data points as follows:  
$$x = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]$$
$$y = [2.56, 1.14, 4.01, 6.02, 4.62, 7.48, 7.79, 9.40, 10.43, 9.23, 13.22, 13.94, 14.30, 15.32, 14.97]$$
- 1.2. Center the dataset (Subtract the mean from each attribute). Plot the datapoints.  
  
**Question 1: What assumption would you make about the dimensionality of the data? (8 points)**
- 1.3. Compute the covariance matrix. Do the features tend to increase(decrease) together?
- 1.4. Now compute the eigenvalues and eigenvectors of the covariance matrix.  
**Question 2: Extract the eigenvector with the highest eigenvalue ( $V$ ). Is there any relation between the direction of  $V$  and direction of the spread of the datapoints? (11 points)**

- 1.5. Now generate a set of random datapoints with 100 samples in a 2-dimensional space. Center the dataset and plot the datapoints.

**Question 3: Compute the covariance matrix, find the eigenvalues and compare them. What would be the true dimensionality of the data? Can you prioritize one of the dimensions? (8 point)**

## 2. Eigenfaces:

- 1) In the next step we are going to deploy the PCA idea to reduce the dimension of a face database. The provided database consists of 400 face images of people in different orientation and lightning. The dimensions of each image is  $112 \times 92$ . Download the Face dataset from eclass and extract the content of “face.zip” into your current Matlab directory. The data consists of two part; training set and testing set. The former is available in the folder named “training”, and the latter is available in “testing” folder. You should extract the eigenfaces of training in this lab and test the performance of a trained model on testing set in the next lab. First, you should load the training set into Matlab. The training set consists of 40 folders, and each folder contains 8 images of a distinct person. So, in total, you have a training face dataset which consists of 320 images of 40 different persons. To load the dataset into Matlab, you should access the folders contained in “training” one by one and load all the images in each subfolder in Matlab.

The following lines of Matlab code displays all the images in training dataset:

```
folders = dir('training');
folders=folders(~ismember({folders.name},{' ','..'}));
subFolders = folders([folders.isdir]);

for k = 1 : length(subFolders)
    cur_dr=['training\' subFolders(k).name];
    images=dir(cur_dr);
    images=images(~ismember({images.name},{' ','..'}));
    for i=1 : length(images)
        imshow(imread([cur_dr \' images(i).name]));
    end
end
```

- 2) Reshape each face data matrix to a vector and arrange all of the samples in a single matrix such that each column corresponds to each face data (as discussed in the lab session). The size of the final matrix should be  $10304(\text{number of features})$  by  $320(\text{number of samples})$
- 3) Now calculate the mean image by averaging through the columns axis. The size of the mean image should be  $10304 \times 1$ .

**Question 4: Reshape the mean image to have the same dimension as each face image and plot it. Save mean image as ‘mean.bmp’, and attach it to the zip file. (8 point)**

- 4) Now subtract the mean image from all of the samples. At the end of this part, the data matrix has been formed ( $X$ ).
- 5) As discussed in the lab session, computing the covariance matrix (a  $10304 \times 10304$  matrix) for the resulted matrix is a tedious procedure. As an alternative method, you’d better form the  $T = \frac{1}{n} X^T X$  and compute its eigenvectors and eigenvalues. The size of the  $T$  matrix should be  $320 \times 320$ . Refer to the lab session slides for more information.

**Question 5: Pick the 5 eigenvectors (eigenfaces) with the largest eigenvalues as the new axes of the low-dimensional space. Reshape the 5 eigenfaces and plot them. Attach the resulted images “Q5\_ithEigen.bmp” to the zip folder (15 points)**

**Hint: Matlab EIG() function returns eigenvalues in ascending order.**

**Question 6: Plot the 5 eigenfaces with the lowest eigenvalues. Is there any information in the plots? Do they look like a generic face image? Attach the resulted images “Q6\_ithEigen.bmp” to the zip folder (12 points)**

### **3. Eigenfaces extraction using SVD decomposition:**

- 1) Now decompose the centered data matrix ( $X$ ) to  $U, D, V$  where  $U$  is the column orthonormal matrix of left singular vectors,  $D$  is the matrix with singular values on diagonal, and  $V$  is the row orthonormal matrix of right singular vectors. As discussed in the lab session, the eigenfaces of the covariance matrix would be the columns of  $U$  and the eigenvalues would be  $\frac{1}{n} * \text{diagonal entries of } D^2$ . (You may use your own code or predefined “svd” function of matlab).

**Question 7: Plot the 5 eigenfaces with the highest eigenvalues again (supposed to see the same results as Question 5). Attach the resulted images “Q7\_ithEigenSVD.bmp” to the zip folder.(10 points)**

**Hint: Matlab SVD() function returns eigenvalues in descending order.**

## 4. Reconstruction:

- 1) At the final step, you are going to reconstruct an original image based on its projection on the 50 eigenfaces with the highest eigenvalues. Choose the first image of the first subject from the training set and compute the contribution of each of the eigenfaces based on the below formula **and the extracted eigenvectors of the SVD decomposition**:

$$W_{50 \times 1} = (V_{10304 \times 50})^T (x_{10304 \times 1} - E\{X\}_{10304 \times 1})$$

- 2) Reconstruct the original image based on the following formula and plot it.

$$\hat{x}_{10304 \times 1} = V_{10304 \times 50} W_{50 \times 1} + E\{X\}_{10304 \times 1}$$

**Question8: Attach the original image and its reconstruction named as ‘Q9org.bmp’ and ‘Q8rec.bmp’, respectively to the zip file. (18 point)**

**Question 9: Now choose 100 eigenfaces. Reconstruct the original image again and plot it. What do you see? How would you choose the optimal number of eigenfaces? Save and attach the result of reconstruction (Q9rec.bmp) to the zip file (10 points)**

**Question 10 (Bonus): Download the Matlab code ‘faceDetect.m’ from eclass. This code takes one image with 112\*92 dimensions. Take 10 photos of yourself with different viewpoints. Add your photos as the images of the 41th person of the face dataset (8 images in training and 2 images in testing). Extract eigen faces for the new dataset and reconstruct one of your images. Attach the original image and its reconstruction named as ‘Q10org.bmp’ and ‘Q10rec.bmp’ to the zip file. (You may use your own face-cropping module instead of using the provided function or do it manually) (10 points)**