Instructions

- Comment your matlab code.
- In your writeup, include your results and a paragraph or two discussing each result. For this homework assignment, do not include the matlab code as part of the writeup.
- Submit a zip file containing your writeup and matlab code.
- Take advantage of the `getAmatrix.m` function.

1 The Yale face database

In this assignment, we will have a look at some simple techniques for object recognition, in particular, we will try to recognize faces. The face data that we will use is derived from the Yale Face Database - for more information, see [http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html](http://cvc.yale.edu/projects/yalefacesB/yalefacesB.html). The database consists of 5760 images of 10 individuals, each under nine poses and 64 different lighting conditions. The availability of such standardized databases is important for scientific research as they provide a common testing ground to test the efficacy of different algorithms.

![Figure 1: The Yale face database B.](image)

In this assignment, we will only use 640 images corresponding to a frontal orientation of the face. These faces are included in the file `yaleBfaces.zip`. You will find the faces divided into five different subsets. Subset 0 consists of images where the light source direction is almost frontal, so that almost all of the face is brightly illuminated. From subset 1 to 4, the light source is progressively moved toward the horizon, so that the effects of shadows increase and not all pixels are illuminated.

The faces in subset 0 will be used as training images, and subsets 1 to 4 will be used as test images.
2 Recognition using Eigenfaces

- Write a function `eigenTrain` which takes as input a matrix $A$ of vectorized images from subset 0. Perform PCA on the data represented by the matrix $A$ and return the top 20 eigenvectors. Note you can use the matlab command `svds` to find the first $k$ eigenvectors. (10 points)

- Rearrange the top 9 eigenvectors you obtained in the previous step into 2D images of size 50 x 50. Display these images by appending them together or using Matlab subplot. (5 points)

- Write a function called `eigenTest` which takes as input a matrix $T$ consisting of vectorized images from subset 0 (the training dataset), a matrix $A$ consisting of vectorized images from one of the test subsets (1-4), the array of eigenvectors from the first step, and a third parameter $k$ that specifies how many eigenvectors to use. Project each image from $A$ onto the space spanned by the first $k$ eigenvectors. Use the $L_2$ distance metric to classify each test image based on its nearest neighbor. To do this, first project the images from the training dataset $T$ to the lower dimensional space. Then, for each image $a_i$ in the test dataset $A$, project it to the lower dimensional space, then find the nearest neighbor $t_j$ in the low dimensional representation of $T$. Give image $a_i$ the same label as $t_j$.

  The output of `eigenTest` should be a the fraction of images from matrix $A$ that were misclassified (error rate) when the image was projected onto the first $k$ eigenvectors. Repeat this experiment for subsets 2 through 5 and for $k = 1..20$. Store these results in a $4 \times 20$ matrix $R$ where $R_{ik}$ is the error rate of subset $i$ using the top $k$ eigenvectors. Plot the error rate of each subset as a function of $k$ in the same plot, use the matlab `legend` function to add a legend to your plot. (10 points)

- Repeat the experiment from the previous step, but throw out the first three eigenvectors. That is, use $k$ eigenvectors starting with the fourth eigenvectors. Produce a plot similar to the one in the previous step. How do you explain the difference in recognition performance from part (c)? (5 points)

- Explain any trends you observe in the variation of error rates as you move from subsets 1 to 4 and as you increase the number of eigenvectors. Use images from each subset to reinforce your claims. (5 points)

3 Recognition using Fisherfaces

- Write a function called `fisherTrain` which takes as input a matrix $A$ of vectorized images from subset 0, and the number of classes $c$. Let the number of training images be $N$ (in this case, $N = 70$) and the desired number of bases be $c - 1$. Your function should:
  - Use PCA to compute the first $N - c$ principal components, let this be $W_{PCA}$.
  - Use $W_{PCA}$ to project the training data into a space of dimension $N - c$.
  - Compute $S_B$ and $S_W$ on the output from the previous step.
  - Compute $W_{fld}$, the generalized eigenvectors for $S_B$ and $S_W$, keeping the $c - 1$ eigenvectors corresponding to the largest eigenvalues.
  - The fisher bases will be a $W_{opt} = W_{fld}^T W_{PCA}^T$.

  The output of your function should be the fisher bases. (10 points)

- As in the Eigenfaces exercise, rearrange the top 9 Fisher bases you obtained in the previous step into 2D arrays of size 50 x 50 and display them. (5 points)
• As in the eigenfaces exercise, perform recognition on the test set with Fisherfaces. Write a function `fisherTest` that takes as input a matrix \( T \) consisting of vectorized images from subset 0 (the training dataset), a matrix \( A \) consisting of vectorized images from one of the test subsets (1-4), the array of Fisher bases, and a parameter \( k \) specifying how many Fisher bases to use. This function should return the error rate for the test set.

Run this experiment for each subset and for \( k = 1 \ldots 9 \). Store your results in a \( 4 \times 9 \) matrix and produce a plot similar to the one you produced in the eigenface exercise. Explain any trends you observe in the variation of error rates with different subsets and different values of \( k \). (10 points)

4 Recognition using linear subspaces

• Write a function called `linearBasis` which takes as input the index of a person \( i \) and returns the 3D linear basis \( b_i \) as a \( 2500 \times 3 \) array. Your function should get the vectorized images of person \( i \) from the training images in subset 0 and arrange them in a \( 2500 \times 7 \) matrix \( M \). Perform SVD on this matrix and return the first three columns of \( U \) where \( M = U \Sigma V^\top \). Store a basis \( b_i \) for each person in the dataset. (10 points)

• As in the previous parts, rearrange the basis vectors of each person and display them. You can arrange these images in a \( 10 \times 3 \) format (3 images for each of 10 persons in the dataset). (5 points)

• As in the previous parts, perform recognition on the test set using the linear subspace method. Write a function `linearBasisTest` which takes as input a matrix \( A \) consisting of vectorized images from one of the subsets, and the set of basis vectors \( b_i \) for each person. This function should return the error rate for the test set. To classify each test image, use the distance to the subspace defined by

\[
d(x, i) = |x - B_i B_i^\top x|,
\]

where \( x \) is the test image and \( B_i \) is the basis for person \( i \). In other words, the label should be \( \text{arg min}_i d(x, i) \). Run this experiment for each subset and store your results in a \( 4 \times 1 \) vector \( R \) where \( R_i \) is the error rate on subset \( i \). Produce a plot of the error rate for each subset using the linear subspace method. (10 points)

• Extra credit: Produce a bar plot comparing the linear subspace method to the other two methods using the value \( k \) that produced the best results in the eigenfaces and fisher faces experiments. Provide a concise summary of the results. (5 points).

5 Writeup

I will be looking for the following in our writeup:

• All images should be clearly labeled and have a short paragraph/caption explaining what the figure is about. (5 points)

• Your document should be properly formatted and organized. (5 points)

• If a question asks you to “explain the results”, then write a paragraph using complete sentences. Try to come up with reasonable explanations for the results you are getting. This may involve doing a little research including material outside of lecture notes and the textbook. It may be possible to earn extra credit if your research includes external sources, but be sure to cite your sources. (5 points)