Mini Project # 3 **Due: Feb. 28/2024**

1. Get the data file allFaces.mat from Canvas (you can read .mat files both in Matlab and Python). Load the file.

```
load allFaces.mat
```

Face pictures of 38 people with several different room lightings are stored in this file. The size of each picture is n=192 by m=168. Each picture is reshaped as a vector of 32256 by 1 and stored as a column in the matrix named faces. When you load the file, you get this matrix and also a vector named nfaces. The vector nfaces is a 1 by 38 row vector. The first element of nfaces shows howmany pictures of person # 1 is stored in the faces matrix. For instance, nfaces(1)=64. This means that from column 1 to column 64 we have pictures of person#1 with 64 different room lightings. Nfaces(2) = 64 as well. This means that in the matrix faces from column 65 to 128 we have 64 pictures of person # 2, and so on. Not all element of nfaces are equal.

Take a look the following few code lines as an example of how you can read these pictures and plot them:

```
A. Reading first pictures of person # 1 to person # 36:

Person = zeros(n,m,36) ;
figure(1)
for i = 1:36
    Person(:,:,i) = reshape(faces(:,1+sum(nfaces(1:i-1))),n,m);
    subplot(6,6,i)
    imagesc(Person(:,:,i)) ; colormap gray ;  axis off ;
    pbaspect([0.8802 1 1])  ;
end
```

Output:

Reading 64 different pictures of Person #1, then Person # 2 ….

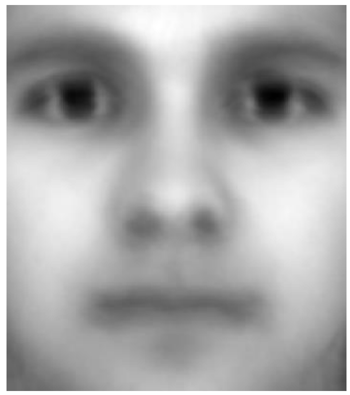
```
figure(2)
Snapshot = zeros(n,m,64) ;
for i = 1:length(nfaces)
   i
   subset = faces(:,1+sum(nfaces(1:i-1)):sum(nfaces(1:i)));
   for j = 1:nfaces(i)
       Snapshot(:,:,j) = reshape(subset(:,j),n,m);
       subplot(6,11,j)
       imagesc(Snapshot(:,:,j)) ; colormap gray ; axis off ;
       pbaspect([0.8802 1 1])  ;
   end
   pause
end
```

Output: Person # 1

Output: Person # 2

2. Load all faces with all different lightings from person # 1 to person # 36. Calculate the average face picture. Subtract that average from all pictures. Store these average subtracted faces in matrix X.

   <u>Output:</u>



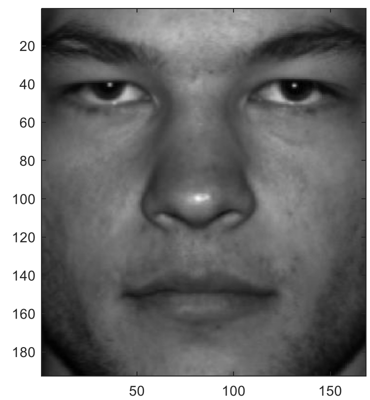
3. Calculate all singular values and singular vectors of X (in Matlab: [U,S,V]=svd(X,’econ’)). Reshape these U vectors to 2-dimensional pictures of size 192 by 168. These are our eigen faces. Plot the first 54 eigen faces.

   <u>Output:</u>

   Carefully investigate these eigen faces. Do you think each eigen face is trying to detect certain features?

Compute the inner product of the first and the 5th eigen faces (do the computations with vectors, before reshaping to 2-dimensional pictures). Are these eigen faces orthogonal? How about the 10th eigen face and 15th eigen face? Are they all orthogonal?

4. Now let's decompose the first picture of person # 37 to these eigen faces.
<u>Output: First picture of person # 37</u>

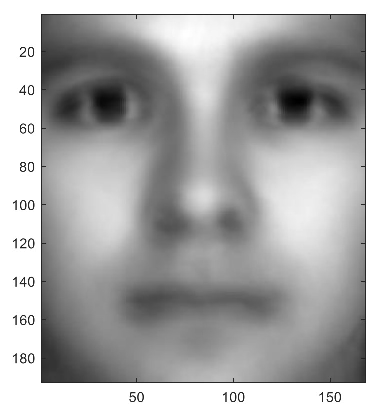


If we assume that vector V is the first picture of person # 37, then:

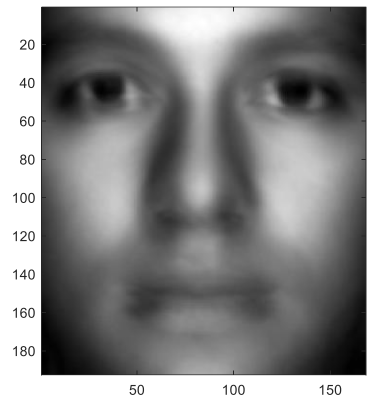
$V \approx \sum_{i=1}^{r} \alpha_i U_i$ where $\alpha_i = (U_i)^T V$

We are truncating at some r. What is the maximum value of r?
Plot the approximation of vector V as a 2-dimensional picture for r=5.

<u>Output: r=5 approximation for the first picture of person # 37</u>

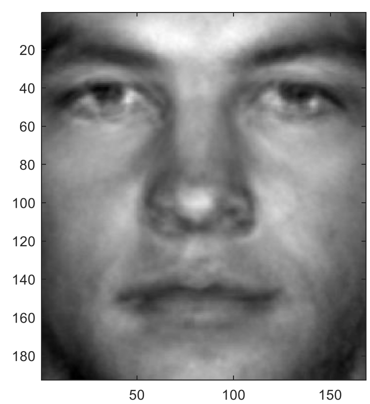


Repeat this for r = 10
<u>Output: r=10 approximation for the first picture of person # 37</u>



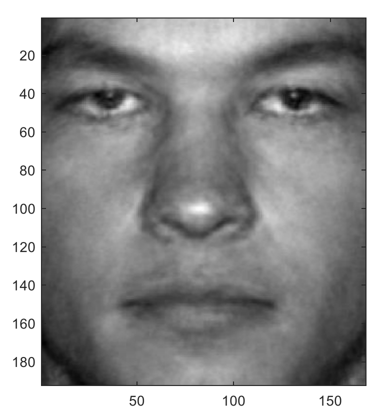
Repeat this for r = 200

<u>Output: r=200 approximation for the first picture of person # 37</u>



Repeat this for r = 800

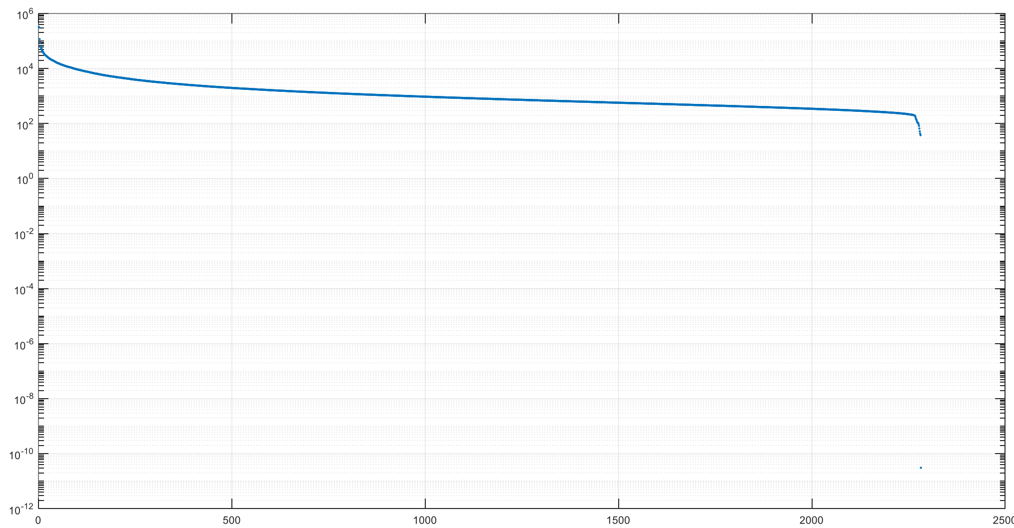
<u>Output: r=800 approximation for the first picture of person # 37</u>



Do you think you can find a good approximation of the picture with r=100?

5. Plot singular values (in a semi-logarithmic scale, horizontal axis representing the index, vertical axis representing the value of the singular value, vertical axis is scaled logarithmically). Do you see a good point for truncation?

<u>Output:</u>



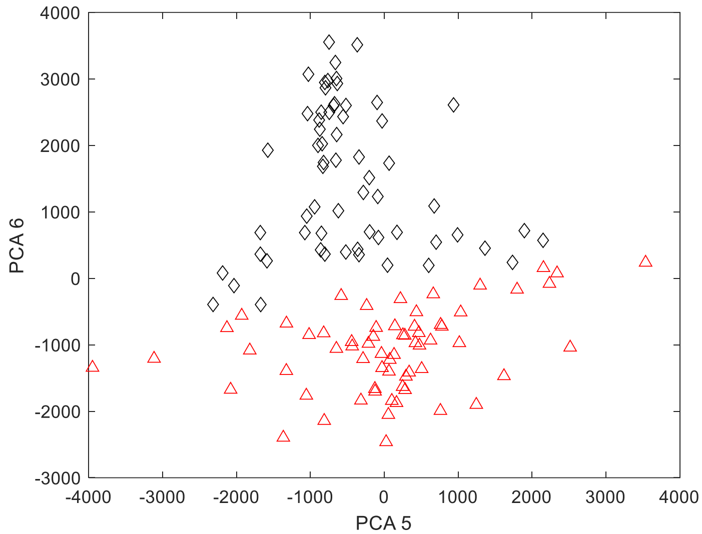
6. Get all 64 different pictures of Person # 2. How much of eigen face number 5 do you have in each picture of person # 2? (For instance, if V_1 is the vector that represents the first picture of person # 2, and $U(5)$ is the 5th eigen face, $\alpha = (V_1)^T\ U(5)$, we have α amount of 5th eigen face in the first picture of person 2).

   How much of each eigen face 5 do we have in every picture of person # 2?

   How much of each eigen face 6 do we have in every picture of person # 2?

   Plot a 2-dimensional graph in which the horizontal axis is the 5th eigen face and vertical axis is the 6th eigen face. In this plane mark much of each eigen faces 5th and 6th exists in all pictures of person 2 (black color) and also person 7 (Red color). Can you use this graph to implement a good classification algorithm for face recognition? Can you do this with SVM?

   <u>Output:</u>


4000
3000
2000
1000
0
-1000
-2000
-3000
PCA 6
-4000
-3000
-2000
-1000
0
1000
2000
3000
4000
PCA 5