AMATH 582 Homework 4

Reed Nomura

March 06, 2020

Abstract

In this paper we will be exploring some applications of singular value decompositions (SVDs). We will be exploring the use of SVDs to categorize data. Specifically, we will be looking at methods for powering machine learning strategies for image and audio recognition.

1 Introduction and Overview

Principal component analysis is a procedure used to transform data into orthogonal dimensions in which our data exists. These dimensions provide important insight into the coordinate system in which our data exists and is best interpreted. In this paper we will be using principal component analysis to understand the distinguishing features of various photos and audio clips. Initially we will be examining several sets of photographs. We will be performing a SVD on the photos in order to determine what makes each subject and expression unique. These methods could be applied to create facial identity or expression recognition software. After this we will be exploring the use of SVDs to classify music. We will be using the SVD of a set of music clips to determine the differentiating features of various bands and genres of music. These methods could be applied to create song, band, or genre recognition software.

2 Theoretical Background

2.1 Singular Value Decomposition (SVD)

A singular value decomposition (SVD) is a factorization of a matrix into a number of constitutive components all of which have a specific meaning in applications.[1]. SVD dcomposition takes on the form

$$A = U\Sigma V^* \tag{1}$$

with

$$U \in \mathbb{C}^{mxm} \text{ is unitary} \tag{2}$$

$$V \in \mathbb{C}^{nxn} \text{ is unitary} \tag{3}$$

$$\Sigma \in \mathbb{R}^{mxn} \text{ is diagonal} \tag{4}$$

2.2 Eigenvector Decomposition

The most straightforward way to diagonalize the covariance matrix is by making the observation that XXT is a square, symmetric m × m matrix, i.e. it is self-adjoint so that the m eigenvalues are real and distinct. [1] Linear algebra provides theorems which state that such a matrix can be rewritten as

$$XX^T = S\Lambda S^{-1} = S\Lambda S^T \tag{5}$$

Instead of working with X, we are able to work with

$$Y = S^T X \tag{6}$$

It then follows that

$$C_Y = \frac{1}{n-1} YY^T \tag{7}$$

$$= \frac{1}{n-1}(S^T X)(S^T X)^T \tag{8}$$

$$= \frac{1}{n-1}(S^T X)(X^T S) \tag{9}$$

$$= \frac{1}{n-1} S^T XX^T S \tag{10}$$

$$= \frac{1}{n-1} S^T \Lambda S^T S \tag{11}$$

Where

$$\Lambda = \begin{bmatrix} \Lambda_1 & & & \\ & \Lambda_2 & & \\ & & \ddots & \\ & & & \Lambda_n \end{bmatrix} \tag{12}$$

Hence

$$C_Y = \frac{1}{n-1}\Lambda \tag{13}$$

In this basis, the principal components are the eigenvectors of XXT with the interpretation that the jth

diagonal value of CY is the variance of X along xj, the jth column of S [1].

2.3 SVD for Diagonalizing Covariance Matrix

A second method for diagonalizing the co- variance matrix is the SVD method. In this case, the SVD can diagonalize any matrix by working in the appropriate pair of bases U and V as outlined in the first lecture of this section. Thus by defining the transformed variable

$$Y = U^* X \tag{14}$$

where U is the unitary transformation associated with the SVD: $X = U\ \Sigma\ V^*$. Just as in the eigenvalue/eigenvector formulation, we then compute the variance in Y:

$$C_Y = \frac{1}{n-1} Y Y^T \tag{15}$$

$$= \frac{1}{n-1} (U^* X)(U^* X)^T \tag{16}$$

$$= \frac{1}{n-1} U^* (X X^T) U \tag{17}$$

$$= \frac{1}{n-1} U^* U \Sigma^2 U U^* \tag{18}$$

$$= \frac{1}{n-1} \Sigma^2 \tag{19}$$

This makes explicit the connection between the SVD and the eigenvalue method, namely that $\Sigma^2 = \Lambda$[1].

2.4 Linear discrimination analysis (LDA)

This is a method for categorizing data through the minimization of dimensions of the data. The goal of LDA is two-fold: find a suitable projection that maximizes the distance between the inter-class data while minimizing the intra-class data [1].

3 Algorithm Implementation and Development

3.1 Picture Outline

1. Load photos
2. Transform photos to vectors
3. Combine data to Matrix
4. Perform SVD on Matrix
5. View Data
6. Compare Specific data vs Total Data to create category criteria
7. Perform LDA to create decision line

3.2 Music Outline

1. Load Music
2. Clip to 5 seconds
3. Create Spectogram from music clip
4. Transform Spectograms into Vector data
5. Put into Data Matrix
6. Perform SVD on Matrix
7. View Data
8. Compare Specific data vs Total Data to create category criteria
9. Perform LDA to create decision line

3.3 Algorithms for Photos

Algorithm 1: Convert Photo Bank to Matrix

```
[m,n] = Photo Size
Set photo count = k
Create Matrix D with dimensions kxm * n
for i = 1 : k do
    Load Photo i
    Reshape Photo i into vector
    Store Photo vector into Row i of D
end for
```

3.4 Algorithms for Music

Algorithm 2: Convert Sound Clips to Matrix

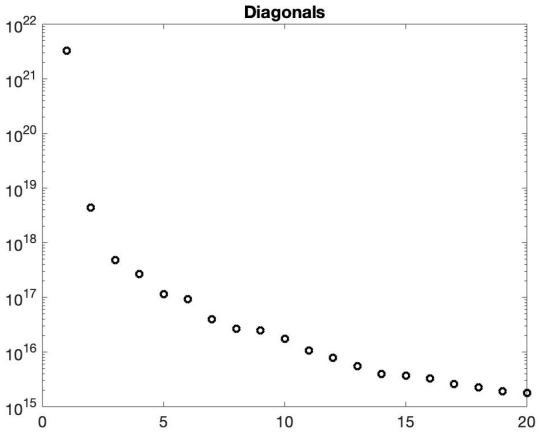
for $i = 1 : k$ **do**
  Load Song i
  Resize Song i to 5 seconds long
  Create Spectogram of song i
  [m,n] = Spectogram Size
  Set Song Count = k
  Reshape Spectogram of song i into vector
  Store Photo vector into Row i of D
end for

4 Computational Results

4.1 Analyzing Faces

In the initial test I analyzed a series of uncropped photos of faces. The data set consisted of 15 photos each of 12 different subjects for a total of 180 photos. Each photo was converted into a vector and placed into a matrix. The SVD of the data matrix revealed the following information.

Plot of the Diagonals



Top Four Dominate Faces

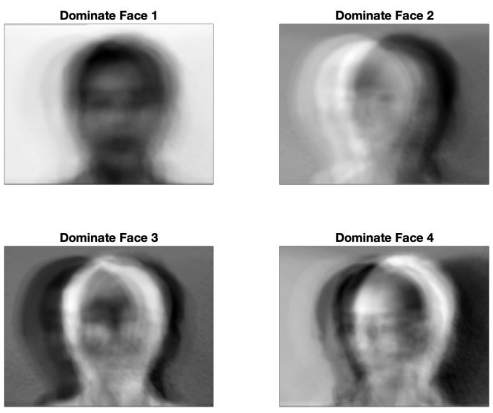
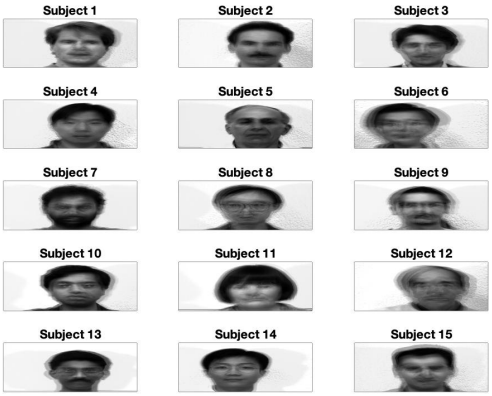


Table 1: Dominate modes for the uncropped images

The diagonals of s are graphed on a semi-log plot on the left of Table 1. This graph reveals that, Since σ_1 is on an order of about 1000 times the size of σ_2, this collection appears to be a rank 1 or possibly rank 2 set. On the right we can see the top four dominate eigenfaces. Since this is a rank 1 or 2 set of data, we are more concerned with the the first two of these faces. In order to develop a method for facial recognition. We need to know how each of our subjects is distinctly different from our eigenfaces.

Each Subject Averaged



Subject Key

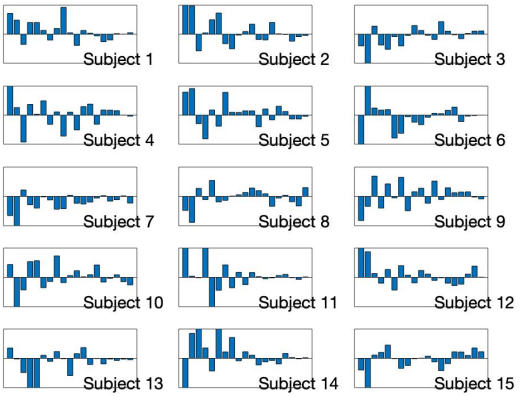
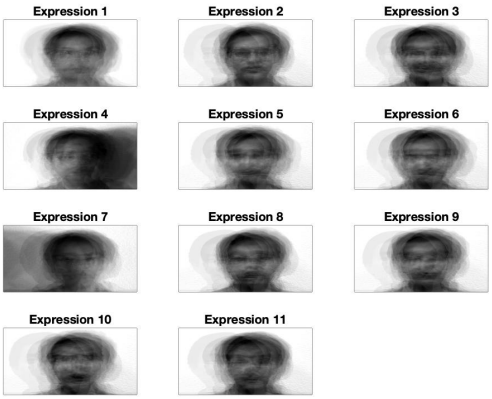


Table 2: Facial recognition for uncropped images

On the left of Table 2 we have the averaged photos for each subject. These photos are then compared against the rank 2 eigenface. On the right of the table we can see the amount that each averaged subject differs from the average face and in which modes the differ. These objects on the right can be used as a key that identifies the distinct ways each subject's face is unique from the rest. We now have a metric by which we can identify a face by pure pixel data. We can repeat this same strategy with the expressions instead of the subjects.

Each Expression Averaged



Expression Key

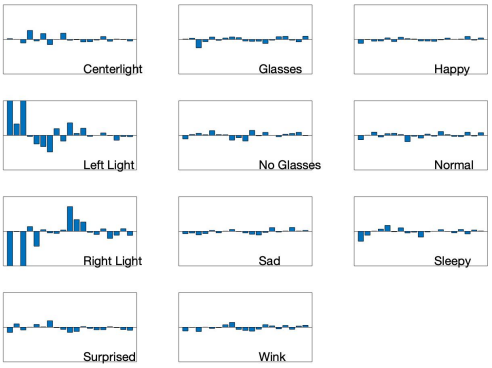
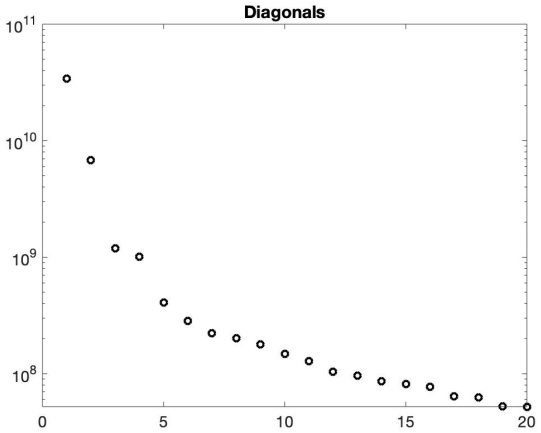


Table 3: Expression recognition for uncropped images

On the left in Table 3 we have the averaged photos of each expression. On the right we can see the key for each expression along with the proper labels for these expressions. This could be used to recognize expressions on peoples faces or, in the case of this experiment, light angles.

We are going to repeat the same basic steps that we just completed with the set of cropped images to compare the results. The data set of cropped images was composed of 39 subjects, each with 64 photos for a total of 2,469 photos. Given the large quantity of data and the time it takes to process it, I decided to only use the first 10 subjects for this case. Each cropped image was uploaded, converted to a vector, and added into a matrix. I performed the SVD on the matrix and viewed the results.

Plot of the Diagonals



Top Four Dominate Faces

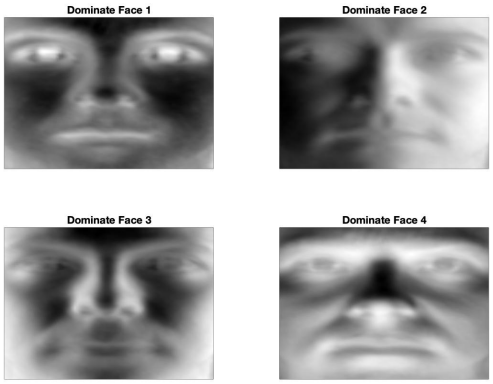
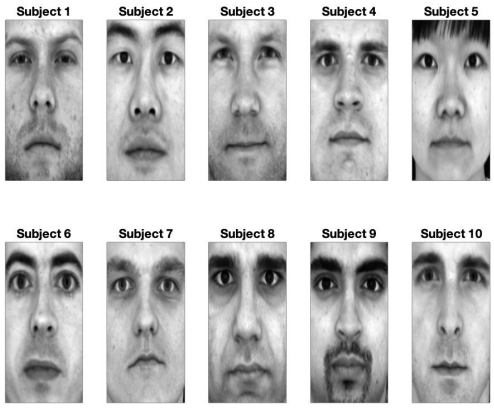


Table 4: Dominate modes for the Cropped images

With the cropped photos there were more modes that were significant. On the left of Table 4 you can that this is at least a rank 2 set of data, if not a rank 4. On the right we can see the first four eigenfaces.

Each Subject Averaged



Subject Key

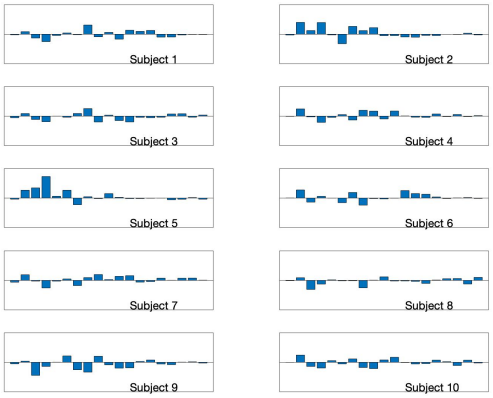


Table 5: Facial Recognition for Cropped Images

Just like with the cropped images, we averaged each subjects' images and compared them against the rank 2 eigenface. On the left it is clear that each of the subjects set of images was much more uniform than in the uncropped cropped case.

4.2 Analyzing Music

In this section we view sets of audio data and determine how to categorize it based on the SVD of the data. The first step was to analyze clips from three different bands and determine if they could be distinguished from one another using SVDs and LDAs. Eight songs were loaded from each other three artists. I chose to include pieces from Yo-Yo-Ma, Eminem, and the Foo Fighters. Their music was loaded and their spectograms created and stored. Each spectogram was converted into a vector and stored into a matrix. The SVD of the matrix was calculated and LDA implemented to categorize the data. Each category proved to be distinct enough to prove the effectiveness of these divisions.

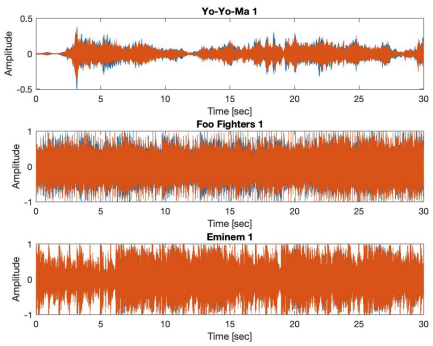


Figure 1: First Look at Three Bands

A preliminary look at the band analysis data in Figure 1 will tell you that these sets are very distinct.

The second test involved music from three bands that share a genre. This will be much more difficult to distinguish. Figure 2 shows how similar these clips can be.

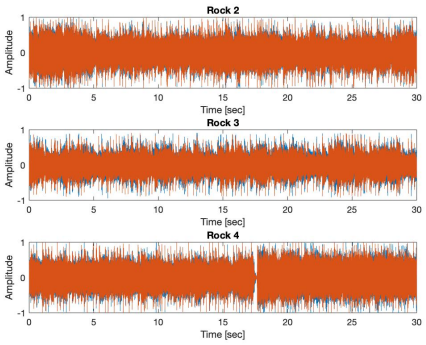


Figure 2: First Look at Rock Genre

The last test involves looking at trying to categorize Genres. These will prove to be easier than categorizing bands within a single genre. But likely, this will be slightly more difficult than categorizing distinct bands.

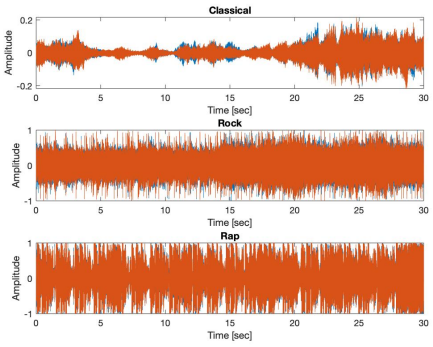


Figure 3: First Look at Genre

5 Summary and Conclusions

Essentially every piece of this paper followed the same basic steps. In each case we formatted the data into a matrix, performed the SVD on the matrix, and used that information to create a key by which we can categorize future data. For the photos we were able to see that the cropped images had more useful information in higher modes and produces more successful category division. In the music analysis we were able to see the value of SVDs in conjunction with LDAs as a means for categorizing audio data. In test 1 the methods produced significant separation of the classes. In test 2 it was much more difficult to distinguish between the bands within the same genre. Finally, test 3 fell somewhere between test 1 and 2 providing moderate separation of classes.

References

[1] Jose Nathan Kutz. *Data-driven modeling & scientific computation: methods for complex systems & big data*. Oxford University Press, 2013.

[2] *MathWorks Website*. URL: `https://www.mathworks.com/help/matlab/index.html`.

Appendix A MATLAB Functions

- `Y = double(X)`: converts the values in X to double precision [2].
- `A = imread(filename)`: reads the image from the file specified by filename, inferring the format of the file from its contents. If filename is a multi-image file, then imread reads the first image in the file [2].
- `s = num2str(A)`: converts a numeric array into a character array that represents the numbers. The output format depends on the magnitudes of the original values. num2str is useful for labeling and titling plots with numeric values[2].
- `pcolor(C)`: creates a pseudocolor plot using the values in matrix C [2].
- `B = flipud(A)`: returns A with its rows flipped in the up-down direction (that is, about a horizontal axis) [2].
- `eval(expression)`: evaluates the MATLAB® code represented by expression. If you use eval within an anonymous function, nested function, or function that contains a nested function, the evaluated expression cannot create a variable. [2].
- `B = reshape(A,sz1,...,szN)`: reshapes A into a sz1-by-...-by-szN array where sz1,...,szN indicates the size of each dimension [2].
- `B = imresize(A,[numrows numcols])`: returns image B that has the number of rows and columns specified by the two-element vector [numrows numcols] [2].
- `size(A)`: returns a row vector whose elements are the lengths of the corresponding dimensions of A. For example, if A is a 3-by-4 matrix, then size(A) returns the vector [3 4].[2]
- `d = eigs(A,k,sigma)`: returns k eigenvalues based on the value of sigma. For example, eigs(A,k,'smallestabs') returns the k smallest magnitude eigenvalues [2].
- `D = diag(v)`: returns a square diagonal matrix with the elements of vector v on the main diagonal. [2].
- `semilogy(Y)`: creates a plot using a base 10 logarithmic scale for the y-axis and a linear scale for the x-axis [2].

- `bar(y)` creates a bar graph with one bar for each element in y. If y is an m-by-n matrix, then bar creates m groups of n bars [2].
- `w = hamming(L)`: returns an L-point symmetric Hamming window [2].

  s = spectrogram(x,window) uses window to divide the signal into segments and perform windowing.
- `[y,Fs] = audioread(filename)`: reads data from the file named filename, and returns sampled data, y, and a sample rate for that data, Fs[2].
- `[M,I] = max()`: returns the index into the operating dimension that corresponds to the maximum value of A for any of the previous syntaxes [2].
- `[row,col] = ind2sub(sz,ind)`: returns the arrays row and col containing the equivalent row and column subscripts corresponding to the linear indices ind for a matrix of size sz. Here sz is a vector with two elements, where sz(1) specifies the number of rows and sz(2) specifies the number of columns [2].
- `B = repmat(A,n)`: returns an array containing n copies of A in the row and column dimensions. The size of B is size(A)*n when A is a matrix [2].
- `sigma = svd(A)`: returns a vector sigma containing the singular values of a symbolic matrix A [2].
- `y = linspace(x1,x2,n)`: generates n points. The spacing between the points is $\frac{x2-x1}{n-1}$ [2]

Appendix B MATLAB Code

This code can be found at: https://github.com/ReedNomura/AMATH-582/blob/master/Homework4.m

```
%AMATH 582 Homework 4
clear all; close all; clc; %Start Fresh
%% Uncropped
%Subject 1
C1_1 = double(imread('subject01.centerlight'));
C2_1 = double(imread('subject01.glasses'));
C3_1 = double(imread('subject01.happy'));
C4_1 = double(imread('subject01.leftlight'));
C5_1 = double(imread('subject01.noglasses'));
C6_1 = double(imread('subject01.normal'));
C7_1 = double(imread('subject01.rightlight'));
C8_1 = double(imread('subject01.sad'));
C9_1 = double(imread('subject01.sleepy'));
C10_1 = double(imread('subject01.surprised'));
C11_1 = double(imread('subject01.wink'));

C_All_1_Ave = (C1_1 + C2_1 + C3_1 + C4_1 + C5_1 + C6_1 + C7_1 + C8_1 + C9_1 + C10_1 ...
    +C11_1)/11;

% Subject 2
C1_2 = double(imread('subject02.centerlight'));
C2_2 = double(imread('subject02.glasses'));
C3_2 = double(imread('subject02.happy'));
C4_2 = double(imread('subject02.leftlight'));
C5_2 = double(imread('subject02.noglasses'));
C6_2 = double(imread('subject02.normal'));
C7_2 = double(imread('subject02.rightlight'));
C8_2 = double(imread('subject02.sad'));
C9_2 = double(imread('subject02.sleepy'));
C10_2 = double(imread('subject02.surprised'));
C11_2 = double(imread('subject02.wink'));

C_All_2_Ave = (C1_2 + C2_2 + C3_2 + C4_2 + C5_2 + C6_2 + C7_2 + C8_2 + C9_2 + C10_2 ...
    +C11_2)/11;
```

```
33
34 %Subject 3
35 C1_3 = double(imread('subject03.centerlight'));
36 C2_3 = double(imread('subject03.glasses'));
37 C3_3 = double(imread('subject03.happy'));
38 C4_3 = double(imread('subject03.leftlight'));
39 C5_3 = double(imread('subject03.noglasses'));
40 C6_3 = double(imread('subject03.normal'));
41 C7_3 = double(imread('subject03.rightlight'));
42 C8_3 = double(imread('subject03.sad'));
43 C9_3 = double(imread('subject03.sleepy'));
44 C10_3 = double(imread('subject03.surprised'));
45 C11_3 = double(imread('subject03.wink'));
46
47 C_All_3_Ave = (C1_3 + C2_3 + C3_3 + C4_3 + C5_3 + C6_3 + C7_3 + C8_3 + C9_3 + C10_3 ...
       +C11_3)/11;
48
49 %Subject 4
50 C1_4 = double(imread('subject04.centerlight'));
51 C2_4 = double(imread('subject04.glasses'));
52 C3_4 = double(imread('subject04.happy'));
53 C4_4 = double(imread('subject04.leftlight'));
54 C5_4 = double(imread('subject04.noglasses'));
55 C6_4 = double(imread('subject04.normal'));
56 C7_4 = double(imread('subject04.rightlight'));
57 C8_4 = double(imread('subject04.normal')); %Corrupted .sad file
58 C9_4 = double(imread('subject04.sleepy'));
59 C10_4 = double(imread('subject04.surprised'));
60 C11_4 = double(imread('subject04.wink'));
61
62 C_All_4_Ave = (C1_4 + C2_4 + C3_4 + C4_4 + C5_4 + C6_4 + C7_4 + C8_4 + C9_4 + C10_4 ...
       +C11_4)/11;
63
64 %Subject 5
65 C1_5 = double(imread('subject05.centerlight'));
66 C2_5 = double(imread('subject05.glasses'));
67 C3_5 = double(imread('subject05.happy'));
68 C4_5 = double(imread('subject05.leftlight'));
69 C5_5 = double(imread('subject05.noglasses'));
70 C6_5 = double(imread('subject05.normal'));
71 C7_5 = double(imread('subject05.rightlight'));
72 C8_5 = double(imread('subject05.sad'));
73 C9_5 = double(imread('subject05.sleepy'));
74 C10_5 = double(imread('subject05.surprised'));
75 C11_5 = double(imread('subject05.wink'));
76
77 C_All_5_Ave = (C1_5 + C2_5 + C3_5 + C4_5 + C5_5 + C6_5 + C7_5 + C8_5 + C9_5 + C10_5 ...
       +C11_5)/11;
78
79 %Subject 6
80 C1_6 = double(imread('subject06.centerlight'));
81 C2_6 = double(imread('subject06.glasses'));
82 C3_6 = double(imread('subject06.happy'));
83 C4_6 = double(imread('subject06.leftlight'));
84 C5_6 = double(imread('subject06.noglasses'));
85 C6_6 = double(imread('subject06.normal'));
86 C7_6 = double(imread('subject06.rightlight'));
87 C8_6 = double(imread('subject06.sad'));
88 C9_6 = double(imread('subject06.sleepy'));
89 C10_6 = double(imread('subject06.surprised'));
90 C11_6 = double(imread('subject06.wink'));
91
92 C_All_6_Ave = (C1_6 + C2_6 + C3_6 + C4_6 + C5_6 + C6_6 + C7_6 + C8_6 + C9_6 + C10_6 ...
       +C11_6)/11;
93
94 %Subject 7
95 C1_7 = double(imread('subject07.centerlight'));
96 C2_7 = double(imread('subject07.glasses'));
```

```
C3_7 = double(imread('subject07.happy'));
C4_7 = double(imread('subject07.leftlight'));
C5_7 = double(imread('subject07.noglasses'));
C6_7 = double(imread('subject07.normal'));
C7_7 = double(imread('subject07.rightlight'));
C8_7 = double(imread('subject07.sad'));
C9_7 = double(imread('subject07.sleepy'));
C10_7 = double(imread('subject07.surprised'));
C11_7 = double(imread('subject07.wink'));

C_All_7_Ave = (C1_7 + C2_7 + C3_7 + C4_7 + C5_7 + C6_7 + C7_7 + C8_7 + C9_7 + C10_7 ...
    +C11_7)/11;

%Subject 8
C1_8 = double(imread('subject08.centerlight'));
C2_8 = double(imread('subject08.glasses'));
C3_8 = double(imread('subject08.happy'));
C4_8 = double(imread('subject08.leftlight'));
C5_8 = double(imread('subject08.noglasses'));
C6_8 = double(imread('subject08.normal'));
C7_8 = double(imread('subject08.rightlight'));
C8_8 = double(imread('subject08.sad'));
C9_8 = double(imread('subject08.sleepy'));
C10_8 = double(imread('subject08.surprised'));
C11_8 = double(imread('subject08.wink'));

C_All_8_Ave = (C1_8 + C2_8 + C3_8 + C4_8 + C5_8 + C6_8 + C7_8 + C8_8 + C9_8 + C10_8 ...
    +C11_8)/11;

%Subject 9
C1_9 = double(imread('subject09.centerlight'));
C2_9 = double(imread('subject09.glasses'));
C3_9 = double(imread('subject09.happy'));
C4_9 = double(imread('subject09.leftlight'));
C5_9 = double(imread('subject09.noglasses'));
C6_9 = double(imread('subject09.normal'));
C7_9 = double(imread('subject09.rightlight'));
C8_9 = double(imread('subject09.sad'));
C9_9 = double(imread('subject09.sleepy'));
C10_9 = double(imread('subject09.surprised'));
C11_9 = double(imread('subject09.wink'));

C_All_9_Ave = (C1_9 + C2_9 + C3_9 + C4_9 + C5_9 + C6_9 + C7_9 + C8_9 + C9_9 + C10_9 ...
    +C11_9)/11;

%Subject 10
C1_10 = double(imread('subject10.centerlight'));
C2_10 = double(imread('subject10.glasses'));
C3_10 = double(imread('subject10.happy'));
C4_10 = double(imread('subject10.leftlight'));
C5_10 = double(imread('subject10.noglasses'));
C6_10 = double(imread('subject10.normal'));
C7_10 = double(imread('subject10.rightlight'));
C8_10 = double(imread('subject10.sad'));
C9_10 = double(imread('subject10.sleepy'));
C10_10 = double(imread('subject10.surprised'));
C11_10 = double(imread('subject10.wink'));

C_All_10_Ave = (C1_10 + C2_10 + C3_10 + C4_10 + C5_10 + C6_10 + C7_10 + C8_10 + C9_10 + ...
    C10_10 +C11_10)/11;

%Subject 11
C1_11 = double(imread('subject11.centerlight'));
C2_11 = double(imread('subject11.glasses'));
C3_11 = double(imread('subject11.happy'));
C4_11 = double(imread('subject11.leftlight'));
C5_11 = double(imread('subject11.noglasses'));
C6_11 = double(imread('subject11.normal'));
```

```
C7_11 = double(imread('subject11.rightlight'));
C8_11 = double(imread('subject11.sad'));
C9_11 = double(imread('subject11.sleepy'));
C10_11 = double(imread('subject11.surprised'));
C11_11 = double(imread('subject11.wink'));

C_All_11_Ave = (C1_11 + C2_11 + C3_11 + C4_11 + C5_11 + C6_11 + C7_11 + C8_11 + C9_11 + ...
    C10_11 +C11_11)/11;

%Subject 12
C1_12 = double(imread('subject12.centerlight'));
C2_12 = double(imread('subject12.glasses'));
C3_12 = double(imread('subject12.happy'));
C4_12 = double(imread('subject12.leftlight'));
C5_12 = double(imread('subject12.noglasses'));
C6_12 = double(imread('subject12.normal'));
C7_12 = double(imread('subject12.rightlight'));
C8_12 = double(imread('subject12.sad'));
C9_12 = double(imread('subject12.sleepy'));
C10_12 = double(imread('subject12.surprised'));
C11_12 = double(imread('subject12.wink'));

C_All_12_Ave = (C1_12 + C2_12 + C3_12 + C4_12 + C5_12 + C6_12 + C7_12 + C8_12 + C9_12 + ...
    C10_12 +C11_12)/11;

%Subject 13
C1_13 = double(imread('subject13.centerlight'));
C2_13 = double(imread('subject13.glasses'));
C3_13 = double(imread('subject13.happy'));
C4_13 = double(imread('subject13.leftlight'));
C5_13 = double(imread('subject13.noglasses'));
C6_13 = double(imread('subject13.normal'));
C7_13 = double(imread('subject13.rightlight'));
C8_13 = double(imread('subject13.sad'));
C9_13 = double(imread('subject13.sleepy'));
C10_13 = double(imread('subject13.surprised'));
C11_13 = double(imread('subject13.wink'));

C_All_13_Ave = (C1_13 + C2_13 + C3_13 + C4_13 + C5_13 + C6_13 + C7_13 + C8_13 + C9_13 + ...
    C10_13 +C11_13)/11;

%Subject 14
C1_14 = double(imread('subject14.centerlight'));
C2_14 = double(imread('subject14.glasses'));
C3_14 = double(imread('subject14.happy'));
C4_14 = double(imread('subject14.leftlight'));
C5_14 = double(imread('subject14.noglasses'));
C6_14 = double(imread('subject14.normal'));
C7_14 = double(imread('subject14.rightlight'));
C8_14 = double(imread('subject14.sad'));
C9_14 = double(imread('subject14.sleepy'));
C10_14 = double(imread('subject14.surprised'));
C11_14 = double(imread('subject14.wink'));

C_All_14_Ave = (C1_14 + C2_14 + C3_14 + C4_14 + C5_14 + C6_14 + C7_14 + C8_14 + C9_14 + ...
    C10_14 +C11_14)/11;

%Subject 15
C1_15 = double(imread('subject15.centerlight'));
C2_15 = double(imread('subject15.glasses'));
C3_15 = double(imread('subject15.happy'));
C4_15 = double(imread('subject15.leftlight'));
C5_15 = double(imread('subject15.noglasses'));
C6_15 = double(imread('subject15.normal'));
C7_15 = double(imread('subject15.rightlight'));
C8_15 = double(imread('subject15.sad'));
C9_15 = double(imread('subject15.sleepy'));
C10_15 = double(imread('subject15.surprised'));
```

```
C11_15 = double(imread('subject15.wink'));

C_All_15_Ave = (C1_15 + C2_15 + C3_15 + C4_15 + C5_15 + C6_15 + C7_15 + C8_15 + C9_15 + ...
    C10_15 +C11_15)/11;

%% Average of all Types
% Centerlight
C_1_All_Ave = (C1_1 + C1_2 + C1_3 + C1_4 + C1_5 + C1_6 + C1_7 + C1_8 + C1_9 + C1_10 +C1_11 ...
    + C1_12 + C1_13 + C1_14 + C1_15)/15;
% Glasses
C_2_All_Ave = (C2_1 + C2_2 + C2_3 + C2_4 + C2_5 + C2_6 + C2_7 + C2_8 + C2_9 + C2_10 +C2_11 ...
    + C2_12 + C2_13 + C2_14 + C2_15)/15;
% Happy
C_3_All_Ave = (C3_1 + C3_2 + C3_3 + C3_4 + C3_5 + C3_6 + C3_7 + C3_8 + C3_9 + C3_10 +C3_11 ...
    + C3_12 + C3_13 + C3_14 + C3_15)/15;
% Leftlight
C_4_All_Ave = (C4_1 + C4_2 + C4_3 + C4_4 + C4_5 + C4_6 + C4_7 + C4_8 + C4_9 + C4_10 +C4_11 ...
    + C4_12 + C4_13 + C4_14 + C4_15)/15;
%N o Glasses
C_5_All_Ave = (C5_1 + C5_2 + C5_3 + C5_4 + C5_5 + C5_6 + C5_7 + C5_8 + C5_9 + C5_10 +C5_11 ...
    + C5_12 + C5_13 + C5_14 + C5_15)/15;
% Normal
C_6_All_Ave = (C6_1 + C6_2 + C6_3 + C6_4 + C6_5 + C6_6 + C6_7 + C6_8 + C6_9 + C6_10 +C6_11 ...
    + C6_12 + C6_13 + C6_14 + C6_15)/15;
% Right Light
C_7_All_Ave = (C7_1 + C7_2 + C7_3 + C7_4 + C7_5 + C7_6 + C7_7 + C7_8 + C7_9 + C7_10 +C7_11 ...
    + C7_12 + C7_13 + C7_14 + C7_15)/15;
% Sad
C_8_All_Ave = (C8_1 + C8_2 + C8_3 + C8_4 + C8_5 + C8_6 + C8_7 + C8_8 + C8_9 + C8_10 +C8_11 ...
    + C8_12 + C8_13 + C8_14 + C8_15)/15;
% Sleepy
C_9_All_Ave = (C9_1 + C9_2 + C9_3 + C9_4 + C9_5 + C9_6 + C9_7 + C9_8 + C9_9 + C9_10 +C9_11 ...
    + C9_12 + C9_13 + C9_14 + C9_15)/15;
% Surprised
C_10_All_Ave = (C10_1 + C10_2 + C10_3 + C10_4 + C10_5 + C10_6 + C10_7 + C10_8 + C10_9 + ...
    C10_10 +C10_11 + C10_12 + C10_13 + C10_14 + C10_15)/15;
% Wink
C_11_All_Ave = (C11_1 + C11_2 + C11_3 + C11_4 + C11_5 + C11_6 + C11_7 + C11_8 + C11_9 + ...
    C11_10 +C11_11 + C11_12 + C11_13 + C11_14 + C11_15)/15;

C_All_All_Ave = (C_1_All_Ave + C_2_All_Ave + C_3_All_Ave + C_4_All_Ave + C_5_All_Ave + ...
    C_6_All_Ave + C_7_All_Ave + C_8_All_Ave + C_9_All_Ave + C_10_All_Ave + C_11_All_Ave)/11;

%% Plot All Faces
figure()
for j = 1:15
for k = 1:11
subplot (15,11,k+(j-1)*11)
 i = (['C',num2str(k),'_', num2str(j)]);
 pcolor(flipud(eval(i))), shading interp, colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
end
end
%% Plot Average of Subjects
figure()
for j = 1:15
subplot (5,3,j)
 i = (['C_All_',num2str(j),'_Ave']);
 k = (['Subject ', num2str(j)]);
 pcolor(flipud(eval(i))), shading interp, colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
 title(k)
end
%% Plot Average of Expressions
figure()
for j = 1:11
subplot (4,3,j)
 i = (['C_',num2str(j),'_All_Ave']);
 k = (['Expression ', num2str(j)]);
 pcolor(flipud(eval(i))), shading interp, colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
```

```
 title(k)
end
%% Average of all Images
figure()
pcolor(flipud(C_All_All_Ave)), shading interp, colormap(gray), set(gca, 'Xtick',[], ...
    'Ytick',[])
 title('Average of All Images')
 %% Create Data Matrix
for jj= 1:15
    for kk = 1:11
 i = (['C',num2str(kk),'-', num2str(jj)]);
D(kk+(jj-1)*11,:) = reshape(imresize(eval(i), [80, 120]), 1, 80*120);
end
end
%% SVD
[u, s, v] = svd(D);
sig = diag(s);
%% Plot Diagonals
figure()
semilogy(sig, 'ko', 'Linewidth', [2])
set (gca, 'Xlim',[0 20], 'Fontsize' , [14])
title('Diagonals')
%% Plot Dominate 4
figure()

subplot(2,2,1), face1 = reshape(v(:,1),80, 120); pcolor(flipud(face1)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 1')
subplot(2,2,2), face2 = reshape(v(:,2),80, 120); pcolor(flipud(face2)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 2')
subplot(2,2,3), face3 = reshape(v(:,3),80, 120); pcolor(flipud(face3)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 3')
subplot(2,2,4), face4 = reshape(v(:,4),80, 120); pcolor(flipud(face4)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 4')

%% Reshape all Subject averages
vecC_1=reshape(imresize(C_All_1_Ave, [80, 120]), 1, 80*120);
vecC_2=reshape(imresize(C_All_2_Ave, [80, 120]), 1, 80*120);
vecC_3=reshape(imresize(C_All_3_Ave, [80, 120]), 1, 80*120);
vecC_4=reshape(imresize(C_All_4_Ave, [80, 120]), 1, 80*120);
vecC_5=reshape(imresize(C_All_5_Ave, [80, 120]), 1, 80*120);
vecC_6=reshape(imresize(C_All_6_Ave, [80, 120]), 1, 80*120);
vecC_7=reshape(imresize(C_All_7_Ave, [80, 120]), 1, 80*120);
vecC_8=reshape(imresize(C_All_8_Ave, [80, 120]), 1, 80*120);
vecC_9=reshape(imresize(C_All_9_Ave, [80, 120]), 1, 80*120);
vecC_10=reshape(imresize(C_All_10_Ave, [80, 120]), 1, 80*120);
vecC_11=reshape(imresize(C_All_11_Ave, [80, 120]), 1, 80*120);
vecC_12=reshape(imresize(C_All_12_Ave, [80, 120]), 1, 80*120);
vecC_13=reshape(imresize(C_All_13_Ave, [80, 120]), 1, 80*120);
vecC_14=reshape(imresize(C_All_14_Ave, [80, 120]), 1, 80*120);
vecC_15=reshape(imresize(C_All_15_Ave, [80, 120]), 1, 80*120);

%% Project Subject averages onto V
figure()
projC_1 = vecC_1*v;
subplot(5,3,1), bar(projC_1(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 1', 'Fontsize', [15])

projC_2 = vecC_2*v;
subplot(5,3,2), bar(projC_2(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 2', 'Fontsize', [15])

projC_3 = vecC_3*v;
```

```
subplot(5,3,3), bar(projC_3(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 3', 'Fontsize', [15])

projC_4 = vecC_4*v;
subplot(5,3,4), bar(projC_4(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 4', 'Fontsize', [15])

projC_5 = vecC_5*v;
subplot(5,3,5), bar(projC_5(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 5', 'Fontsize', [15])

projC_6 = vecC_6*v;
subplot(5,3,6), bar(projC_6(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 6', 'Fontsize', [15])

projC_7 = vecC_7*v;
subplot(5,3,7), bar(projC_7(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 7', 'Fontsize', [15])

projC_8 = vecC_8*v;
subplot(5,3,8), bar(projC_8(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 8', 'Fontsize', [15])

projC_9 = vecC_9*v;
subplot(5,3,9), bar(projC_9(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 9', 'Fontsize', [15])

projC_10 = vecC_10*v;
subplot(5,3,10), bar(projC_10(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 10', 'Fontsize', [15])

projC_11 = vecC_11*v;
subplot(5,3,11), bar(projC_11(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 11', 'Fontsize', [15])

projC_12 = vecC_12*v;
subplot(5,3,12), bar(projC_12(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 12', 'Fontsize', [15])

projC_13 = vecC_13*v;
subplot(5,3,13), bar(projC_13(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 13', 'Fontsize', [15])

projC_14 = vecC_14*v;
subplot(5,3,14), bar(projC_14(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 14', 'Fontsize', [15])

projC_15 = vecC_15*v;
subplot(5,3,15), bar(projC_15(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 15', 'Fontsize', [15])


%% Eigenvalue Decomposition
A = (D')*D;
size(A)
```

```
[V,D] = eigs(A,20, 'lm');
%% Plot Diagonals
figure()
semilogy(diag(D), 'ko', 'Linewidth', [2])
set (gca, 'Fontsize' , [14])
title('Diagonals')
%% Plot Dominate 4
figure()

subplot(2,2,1), face1 = reshape(V(:,1),80, 120); pcolor(flipud(face1)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 1')
subplot(2,2,2), face2 = reshape(V(:,2),80, 120); pcolor(flipud(face2)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 2')
subplot(2,2,3), face3 = reshape(V(:,3),80, 120); pcolor(flipud(face3)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 3')
subplot(2,2,4), face4 = reshape(V(:,4),80, 120); pcolor(flipud(face4)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 4')
%% Reshape all Subject averages
vecC_1=reshape(imresize(C_All_1_Ave, [80, 120]), 1, 80*120);
vecC_2=reshape(imresize(C_All_2_Ave, [80, 120]), 1, 80*120);
vecC_3=reshape(imresize(C_All_3_Ave, [80, 120]), 1, 80*120);
vecC_4=reshape(imresize(C_All_4_Ave, [80, 120]), 1, 80*120);
vecC_5=reshape(imresize(C_All_5_Ave, [80, 120]), 1, 80*120);
vecC_6=reshape(imresize(C_All_6_Ave, [80, 120]), 1, 80*120);
vecC_7=reshape(imresize(C_All_7_Ave, [80, 120]), 1, 80*120);
vecC_8=reshape(imresize(C_All_8_Ave, [80, 120]), 1, 80*120);
vecC_9=reshape(imresize(C_All_9_Ave, [80, 120]), 1, 80*120);
vecC_10=reshape(imresize(C_All_10_Ave, [80, 120]), 1, 80*120);
vecC_11=reshape(imresize(C_All_11_Ave, [80, 120]), 1, 80*120);
vecC_12=reshape(imresize(C_All_12_Ave, [80, 120]), 1, 80*120);
vecC_13=reshape(imresize(C_All_13_Ave, [80, 120]), 1, 80*120);
vecC_14=reshape(imresize(C_All_14_Ave, [80, 120]), 1, 80*120);
vecC_15=reshape(imresize(C_All_15_Ave, [80, 120]), 1, 80*120);

%% Project Subject averages onto V
figure()
projC_1 = vecC_1*V;
subplot(5,3,1), bar(projC_1(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 1', 'Fontsize', [15])

projC_2 = vecC_2*V;
subplot(5,3,2), bar(projC_2(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 2', 'Fontsize', [15])

projC_3 = vecC_3*V;
subplot(5,3,3), bar(projC_3(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 3', 'Fontsize', [15])

projC_4 = vecC_4*V;
subplot(5,3,4), bar(projC_4(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 4', 'Fontsize', [15])

projC_5 = vecC_5*V;
subplot(5,3,5), bar(projC_5(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 5', 'Fontsize', [15])

projC_6 = vecC_6*V;
subplot(5,3,6), bar(projC_6(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
```

```
text( 12, -1700, 'Subject 6', 'Fontsize', [15])

projC_7 = vecC_7*V;
subplot(5,3,7), bar(projC_7(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 7', 'Fontsize', [15])

projC_8 = vecC_8*V;
subplot(5,3,8), bar(projC_8(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 8', 'Fontsize', [15])

projC_9 = vecC_9*V;
subplot(5,3,9), bar(projC_9(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 9', 'Fontsize', [15])

projC_10 = vecC_10*V;
subplot(5,3,10), bar(projC_10(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 10', 'Fontsize', [15])

projC_11 = vecC_11*V;
subplot(5,3,11), bar(projC_11(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 11', 'Fontsize', [15])

projC_12 = vecC_12*V;
subplot(5,3,12), bar(projC_12(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 12', 'Fontsize', [15])

projC_13 = vecC_13*V;
subplot(5,3,13), bar(projC_13(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 13', 'Fontsize', [15])

projC_14 = vecC_14*V;
subplot(5,3,14), bar(projC_14(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 14', 'Fontsize', [15])

projC_15 = vecC_15*V;
subplot(5,3,15), bar(projC_15(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 15', 'Fontsize', [15])

%% Reshape all Expression averages
vecC_1=reshape(imresize(C_1_All_Ave, [80, 120]), 1, 80*120);
vecC_2=reshape(imresize(C_2_All_Ave, [80, 120]), 1, 80*120);
vecC_3=reshape(imresize(C_3_All_Ave, [80, 120]), 1, 80*120);
vecC_4=reshape(imresize(C_4_All_Ave, [80, 120]), 1, 80*120);
vecC_5=reshape(imresize(C_5_All_Ave, [80, 120]), 1, 80*120);
vecC_6=reshape(imresize(C_6_All_Ave, [80, 120]), 1, 80*120);
vecC_7=reshape(imresize(C_7_All_Ave, [80, 120]), 1, 80*120);
vecC_8=reshape(imresize(C_8_All_Ave, [80, 120]), 1, 80*120);
vecC_9=reshape(imresize(C_9_All_Ave, [80, 120]), 1, 80*120);
vecC_10=reshape(imresize(C_10_All_Ave, [80, 120]), 1, 80*120);
vecC_11=reshape(imresize(C_11_All_Ave, [80, 120]), 1, 80*120);

%% Project Expression averages onto V
figure()
projC_1 = vecC_1*V;
subplot(4,3,1), bar(projC_1(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Centerlight', 'Fontsize', [15])

projC_2 = vecC_2*V;
```

```
subplot(4,3,2), bar(projC_2(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Glasses', 'Fontsize', [15])

projC_3 = vecC_3*V;
subplot(4,3,3), bar(projC_3(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Happy', 'Fontsize', [15])

projC_4 = vecC_4*V;
subplot(4,3,4), bar(projC_4(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Left Light', 'Fontsize', [15])

projC_5 = vecC_5*V;
subplot(4,3,5), bar(projC_5(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'No Glasses', 'Fontsize', [15])

projC_6 = vecC_6*V;
subplot(4,3,6), bar(projC_6(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Normal', 'Fontsize', [15])

projC_7 = vecC_7*V;
subplot(4,3,7), bar(projC_7(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Right Light', 'Fontsize', [15])

projC_8 = vecC_8*V;
subplot(4,3,8), bar(projC_8(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Sad', 'Fontsize', [15])

projC_9 = vecC_9*V;
subplot(4,3,9), bar(projC_9(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Sleepy', 'Fontsize', [15])

projC_10 = vecC_10*V;
subplot(4,3,10), bar(projC_10(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Surprised', 'Fontsize', [15])

projC_11 = vecC_11*V;
subplot(4,3,11), bar(projC_11(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Wink', 'Fontsize', [15])


%%AMATH 582 Homework 4 Cropped
clear all; close all; clc; %Start Fresh

%% Cropped
%Subject 1
B01_1 = double(imread('yaleB01_P00A-005E-10.pgm'));
B01_2 = double(imread('yaleB01_P00A-005E+10.pgm'));
B01_3 = double(imread('yaleB01_P00A-010E-20.pgm'));
B01_4 = double(imread('yaleB01_P00A-010E+00.pgm'));
B01_5 = double(imread('yaleB01_P00A-015E+20.pgm'));
B01_6 = double(imread('yaleB01_P00A-020E-10.pgm'));
B01_7 = double(imread('yaleB01_P00A-020E-40.pgm'));
B01_8 = double(imread('yaleB01_P00A-020E+10.pgm'));
B01_9 = double(imread('yaleB01_P00A-025E+00.pgm'));
B01_10 = double(imread('yaleB01_P00A-035E-20.pgm'));
B01_11 = double(imread('yaleB01_P00A-035E+15.pgm'));
B01_12 = double(imread('yaleB01_P00A-035E+40.pgm'));
B01_13 = double(imread('yaleB01_P00A-035E+65.pgm'));
```

```
B01_14 = double(imread('yaleB01_P00A-050E-40.pgm'));
B01_15 = double(imread('yaleB01_P00A-050E+00.pgm'));
B01_16 = double(imread('yaleB01_P00A-060E-20.pgm'));
B01_17 = double(imread('yaleB01_P00A-060E+20.pgm'));
B01_18 = double(imread('yaleB01_P00A-070E-35.pgm'));
B01_19 = double(imread('yaleB01_P00A-070E+00.pgm'));
B01_20 = double(imread('yaleB01_P00A-070E+45.pgm'));
B01_21 = double(imread('yaleB01_P00A-085E-20.pgm'));
B01_22 = double(imread('yaleB01_P00A-085E+20.pgm'));
B01_23 = double(imread('yaleB01_P00A-095E+00.pgm'));
B01_24 = double(imread('yaleB01_P00A-110E-20.pgm'));
B01_25 = double(imread('yaleB01_P00A-110E+15.pgm'));
B01_26 = double(imread('yaleB01_P00A-110E+40.pgm'));
B01_27 = double(imread('yaleB01_P00A-110E+65.pgm'));
B01_28 = double(imread('yaleB01_P00A-120E+00.pgm'));
B01_29 = double(imread('yaleB01_P00A-130E+20.pgm'));
B01_30 = double(imread('yaleB01_P00A+000E-20.pgm'));
B01_31 = double(imread('yaleB01_P00A+000E-35.pgm'));
B01_32 = double(imread('yaleB01_P00A+000E+00.pgm'));
B01_33 = double(imread('yaleB01_P00A+000E+20.pgm'));
B01_34 = double(imread('yaleB01_P00A+000E+45.pgm'));
B01_35 = double(imread('yaleB01_P00A+000E+90.pgm'));
B01_36 = double(imread('yaleB01_P00A+005E-10.pgm'));
B01_37 = double(imread('yaleB01_P00A+005E+10.pgm'));
B01_38 = double(imread('yaleB01_P00A+010E-20.pgm'));
B01_39 = double(imread('yaleB01_P00A+010E+00.pgm'));
B01_40 = double(imread('yaleB01_P00A+015E+20.pgm'));
B01_41 = double(imread('yaleB01_P00A+020E-10.pgm'));
B01_42 = double(imread('yaleB01_P00A+020E-40.pgm'));
B01_43 = double(imread('yaleB01_P00A+020E+10.pgm'));
B01_44 = double(imread('yaleB01_P00A+025E+00.pgm'));
B01_45 = double(imread('yaleB01_P00A+035E-20.pgm'));
B01_46 = double(imread('yaleB01_P00A+035E+15.pgm'));
B01_47 = double(imread('yaleB01_P00A+035E+40.pgm'));
B01_48 = double(imread('yaleB01_P00A+035E+65.pgm'));
B01_49 = double(imread('yaleB01_P00A+050E-40.pgm'));
B01_50 = double(imread('yaleB01_P00A+050E+00.pgm'));
B01_51 = double(imread('yaleB01_P00A+060E-20.pgm'));
B01_52 = double(imread('yaleB01_P00A+060E+20.pgm'));
B01_53 = double(imread('yaleB01_P00A+070E-35.pgm'));
B01_54 = double(imread('yaleB01_P00A+070E+00.pgm'));
B01_55 = double(imread('yaleB01_P00A+070E+45.pgm'));
B01_56 = double(imread('yaleB01_P00A+085E-20.pgm'));
B01_57 = double(imread('yaleB01_P00A+085E+20.pgm'));
B01_58 = double(imread('yaleB01_P00A+095E+00.pgm'));
B01_59 = double(imread('yaleB01_P00A+110E-20.pgm'));
B01_60 = double(imread('yaleB01_P00A+110E+15.pgm'));
B01_61 = double(imread('yaleB01_P00A+110E+40.pgm'));
B01_62 = double(imread('yaleB01_P00A+110E+65.pgm'));
B01_63 = double(imread('yaleB01_P00A+120E+00.pgm'));
B01_64 = double(imread('yaleB01_P00A+130E+20.pgm'));

B01_Ave = (B01_1 + B01_2 + B01_3 + B01_4 + B01_5 + B01_6 + B01_7 + B01_8 + B01_9 + B01_10 ...
    + B01_11 + B01_12 + B01_13 + B01_14 + B01_15 + B01_16 + B01_17 + B01_18 + B01_19 + ...
    B01_20 + B01_21 + B01_22 + B01_23 + B01_24 + B01_25 + B01_26 + B01_27 + B01_28 + ...
    B01_29 + B01_30 + B01_31 + B01_32 + B01_33 + B01_34 + B01_35 + B01_36 + B01_37 + ...
    B01_38 + B01_39 + B01_40 + B01_41 + B01_42 + B01_43 + B01_44 + B01_45 + B01_46 + ...
    B01_47 + B01_48 + B01_49+ B01_50 + B01_51 + B01_52 + B01_53 + B01_54 + B01_55 + B01_56 ...
    + B01_57 + B01_58 + B01_59 + B01_60 + B01_61 + B01_62 + B01_63 + B01_64)/64;

% Subject 2
B02_1 = double(imread('yaleB02_P00A-005E-10.pgm'));
B02_2 = double(imread('yaleB02_P00A-005E+10.pgm'));
B02_3 = double(imread('yaleB02_P00A-010E-20.pgm'));
B02_4 = double(imread('yaleB02_P00A-010E+00.pgm'));
B02_5 = double(imread('yaleB02_P00A-015E+20.pgm'));
B02_6 = double(imread('yaleB02_P00A-020E-10.pgm'));
B02_7 = double(imread('yaleB02_P00A-020E-40.pgm'));
```

```
B02_8 = double(imread('yaleB02_P00A-020E+10.pgm'));
B02_9 = double(imread('yaleB02_P00A-025E+00.pgm'));
B02_10 = double(imread('yaleB02_P00A-035E-20.pgm'));
B02_11 = double(imread('yaleB02_P00A-035E+15.pgm'));
B02_12 = double(imread('yaleB02_P00A-035E+40.pgm'));
B02_13 = double(imread('yaleB02_P00A-035E+65.pgm'));
B02_14 = double(imread('yaleB02_P00A-050E-40.pgm'));
B02_15 = double(imread('yaleB02_P00A-050E+00.pgm'));
B02_16 = double(imread('yaleB02_P00A-060E-20.pgm'));
B02_17 = double(imread('yaleB02_P00A-060E+20.pgm'));
B02_18 = double(imread('yaleB02_P00A-070E-35.pgm'));
B02_19 = double(imread('yaleB02_P00A-070E+00.pgm'));
B02_20 = double(imread('yaleB02_P00A-070E+45.pgm'));
B02_21 = double(imread('yaleB02_P00A-085E-20.pgm'));
B02_22 = double(imread('yaleB02_P00A-085E+20.pgm'));
B02_23 = double(imread('yaleB02_P00A-095E+00.pgm'));
B02_24 = double(imread('yaleB02_P00A-110E-20.pgm'));
B02_25 = double(imread('yaleB02_P00A-110E+15.pgm'));
B02_26 = double(imread('yaleB02_P00A-110E+40.pgm'));
B02_27 = double(imread('yaleB02_P00A-110E+65.pgm'));
B02_28 = double(imread('yaleB02_P00A-120E+00.pgm'));
B02_29 = double(imread('yaleB02_P00A-130E+20.pgm'));
B02_30 = double(imread('yaleB02_P00A+000E-20.pgm'));
B02_31 = double(imread('yaleB02_P00A+000E-35.pgm'));
B02_32 = double(imread('yaleB02_P00A+000E+00.pgm'));
B02_33 = double(imread('yaleB02_P00A+000E+20.pgm'));
B02_34 = double(imread('yaleB02_P00A+000E+45.pgm'));
B02_35 = double(imread('yaleB02_P00A+000E+90.pgm'));
B02_36 = double(imread('yaleB02_P00A+005E-10.pgm'));
B02_37 = double(imread('yaleB02_P00A+005E+10.pgm'));
B02_38 = double(imread('yaleB02_P00A+010E-20.pgm'));
B02_39 = double(imread('yaleB02_P00A+010E+00.pgm'));
B02_40 = double(imread('yaleB02_P00A+015E+20.pgm'));
B02_41 = double(imread('yaleB02_P00A+020E-10.pgm'));
B02_42 = double(imread('yaleB02_P00A+020E-40.pgm'));
B02_43 = double(imread('yaleB02_P00A+020E+10.pgm'));
B02_44 = double(imread('yaleB02_P00A+025E+00.pgm'));
B02_45 = double(imread('yaleB02_P00A+035E-20.pgm'));
B02_46 = double(imread('yaleB02_P00A+035E+15.pgm'));
B02_47 = double(imread('yaleB02_P00A+035E+40.pgm'));
B02_48 = double(imread('yaleB02_P00A+035E+65.pgm'));
B02_49 = double(imread('yaleB02_P00A+050E-40.pgm'));
B02_50 = double(imread('yaleB02_P00A+050E+00.pgm'));
B02_51 = double(imread('yaleB02_P00A+060E-20.pgm'));
B02_52 = double(imread('yaleB02_P00A+060E+20.pgm'));
B02_53 = double(imread('yaleB02_P00A+070E-35.pgm'));
B02_54 = double(imread('yaleB02_P00A+070E+00.pgm'));
B02_55 = double(imread('yaleB02_P00A+070E+45.pgm'));
B02_56 = double(imread('yaleB02_P00A+085E-20.pgm'));
B02_57 = double(imread('yaleB02_P00A+085E+20.pgm'));
B02_58 = double(imread('yaleB02_P00A+095E+00.pgm'));
B02_59 = double(imread('yaleB02_P00A+110E-20.pgm'));
B02_60 = double(imread('yaleB02_P00A+110E+15.pgm'));
B02_61 = double(imread('yaleB02_P00A+110E+40.pgm'));
B02_62 = double(imread('yaleB02_P00A+110E+65.pgm'));
B02_63 = double(imread('yaleB02_P00A+120E+00.pgm'));
B02_64 = double(imread('yaleB02_P00A+130E+20.pgm'));

B02_Ave = (B02_1 + B02_2 + B02_3 + B02_4 + B02_5 + B02_6 + B02_7 + B02_8 + B02_9 + B02_10 ...
    + B02_11 + B02_12 + B02_13 + B02_14 + B02_15 + B02_16 + B02_17 + B02_18 + B02_19 + ...
    B02_20 + B02_21 + B02_22 + B02_23 + B02_24 + B02_25 + B02_26 + B02_27 + B02_28 + ...
    B02_29 +  B02_30 + B02_31 + B02_32 + B02_33 + B02_34 + B02_35 + B02_36 + B02_37 + ...
    B02_38 + B02_39 + B02_40 + B02_41 + B02_42 + B02_43 + B02_44 + B02_45 + B02_46 + ...
    B02_47 + B02_48 + B02_49+ B02_50 + B02_51 + B02_52 + B02_53 + B02_54 + B02_55 + B02_56 ...
    + B02_57 + B02_58 + B02_59 + B02_60 + B02_61 + B02_62 + B02_63 + B02_64)/64;

% Subject 3
B03_1 = double(imread('yaleB03_P00A-005E-10.pgm'));
```

```
694 B03_2 = double(imread('yaleB03_P00A-005E+10.pgm'));
695 B03_3 = double(imread('yaleB03_P00A-010E-20.pgm'));
696 B03_4 = double(imread('yaleB03_P00A-010E+00.pgm'));
697 B03_5 = double(imread('yaleB03_P00A-015E+20.pgm'));
698 B03_6 = double(imread('yaleB03_P00A-020E-10.pgm'));
699 B03_7 = double(imread('yaleB03_P00A-020E-40.pgm'));
700 B03_8 = double(imread('yaleB03_P00A-020E+10.pgm'));
701 B03_9 = double(imread('yaleB03_P00A-025E+00.pgm'));
702 B03_10 = double(imread('yaleB03_P00A-035E-20.pgm'));
703 B03_11 = double(imread('yaleB03_P00A-035E+15.pgm'));
704 B03_12 = double(imread('yaleB03_P00A-035E+40.pgm'));
705 B03_13 = double(imread('yaleB03_P00A-035E+65.pgm'));
706 B03_14 = double(imread('yaleB03_P00A-050E-40.pgm'));
707 B03_15 = double(imread('yaleB03_P00A-050E+00.pgm'));
708 B03_16 = double(imread('yaleB03_P00A-060E-20.pgm'));
709 B03_17 = double(imread('yaleB03_P00A-060E+20.pgm'));
710 B03_18 = double(imread('yaleB03_P00A-070E-35.pgm'));
711 B03_19 = double(imread('yaleB03_P00A-070E+00.pgm'));
712 B03_20 = double(imread('yaleB03_P00A-070E+45.pgm'));
713 B03_21 = double(imread('yaleB03_P00A-085E-20.pgm'));
714 B03_22 = double(imread('yaleB03_P00A-085E+20.pgm'));
715 B03_23 = double(imread('yaleB03_P00A-095E+00.pgm'));
716 B03_24 = double(imread('yaleB03_P00A-110E-20.pgm'));
717 B03_25 = double(imread('yaleB03_P00A-110E+15.pgm'));
718 B03_26 = double(imread('yaleB03_P00A-110E+40.pgm'));
719 B03_27 = double(imread('yaleB03_P00A-110E+65.pgm'));
720 B03_28 = double(imread('yaleB03_P00A-120E+00.pgm'));
721 B03_29 = double(imread('yaleB03_P00A-130E+20.pgm'));
722 B03_30 = double(imread('yaleB03_P00A+000E-20.pgm'));
723 B03_31 = double(imread('yaleB03_P00A+000E-35.pgm'));
724 B03_32 = double(imread('yaleB03_P00A+000E+00.pgm'));
725 B03_33 = double(imread('yaleB03_P00A+000E+20.pgm'));
726 B03_34 = double(imread('yaleB03_P00A+000E+45.pgm'));
727 B03_35 = double(imread('yaleB03_P00A+000E+90.pgm'));
728 B03_36 = double(imread('yaleB03_P00A+005E-10.pgm'));
729 B03_37 = double(imread('yaleB03_P00A+005E+10.pgm'));
730 B03_38 = double(imread('yaleB03_P00A+010E-20.pgm'));
731 B03_39 = double(imread('yaleB03_P00A+010E+00.pgm'));
732 B03_40 = double(imread('yaleB03_P00A+015E+20.pgm'));
733 B03_41 = double(imread('yaleB03_P00A+020E-10.pgm'));
734 B03_42 = double(imread('yaleB03_P00A+020E-40.pgm'));
735 B03_43 = double(imread('yaleB03_P00A+020E+10.pgm'));
736 B03_44 = double(imread('yaleB03_P00A+025E+00.pgm'));
737 B03_45 = double(imread('yaleB03_P00A+035E-20.pgm'));
738 B03_46 = double(imread('yaleB03_P00A+035E+15.pgm'));
739 B03_47 = double(imread('yaleB03_P00A+035E+40.pgm'));
740 B03_48 = double(imread('yaleB03_P00A+035E+65.pgm'));
741 B03_49 = double(imread('yaleB03_P00A+050E-40.pgm'));
742 B03_50 = double(imread('yaleB03_P00A+050E+00.pgm'));
743 B03_51 = double(imread('yaleB03_P00A+060E-20.pgm'));
744 B03_52 = double(imread('yaleB03_P00A+060E+20.pgm'));
745 B03_53 = double(imread('yaleB03_P00A+070E-35.pgm'));
746 B03_54 = double(imread('yaleB03_P00A+070E+00.pgm'));
747 B03_55 = double(imread('yaleB03_P00A+070E+45.pgm'));
748 B03_56 = double(imread('yaleB03_P00A+085E-20.pgm'));
749 B03_57 = double(imread('yaleB03_P00A+085E+20.pgm'));
750 B03_58 = double(imread('yaleB03_P00A+095E+00.pgm'));
751 B03_59 = double(imread('yaleB03_P00A+110E-20.pgm'));
752 B03_60 = double(imread('yaleB03_P00A+110E+15.pgm'));
753 B03_61 = double(imread('yaleB03_P00A+110E+40.pgm'));
754 B03_62 = double(imread('yaleB03_P00A+110E+65.pgm'));
755 B03_63 = double(imread('yaleB03_P00A+120E+00.pgm'));
756 B03_64 = double(imread('yaleB03_P00A+130E+20.pgm'));
757
758 B03_Ave = (B03_1 + B03_2 + B03_3 + B03_4 + B03_5 + B03_6 + B03_7 + B03_8 + B03_9 + B03_10 ...
        + B03_11 + B03_12 + B03_13 + B03_14 + B03_15 + B03_16 + B03_17 + B03_18 + B03_19 + ...
        B03_20 + B03_21 + B03_22 + B03_23 + B03_24 + B03_25 + B03_26 + B03_27 + B03_28 + ...
        B03_29 +  B03_30 + B03_31 + B03_32 + B03_33 + B03_34 + B03_35 + B03_36 + B03_37 + ...
```

```
        B03_38 + B03_39 + B03_40 + B03_41 + B03_42 + B03_43 + B03_44 + B03_45 + B03_46 + ...
        B03_47 + B03_48 + B03_49+ B03_50 + B03_51 + B03_52 + B03_53 + B03_54 + B03_55 + B03_56 ...
        + B03_57 + B03_58 + B03_59 + B03_60 + B03_61 + B03_62 + B03_63 + B03_64)/64;

% Subject 4
B04_1 = double(imread('yaleB04_P00A-005E-10.pgm'));
B04_2 = double(imread('yaleB04_P00A-005E+10.pgm'));
B04_3 = double(imread('yaleB04_P00A-010E-20.pgm'));
B04_4 = double(imread('yaleB04_P00A-010E+00.pgm'));
B04_5 = double(imread('yaleB04_P00A-015E+20.pgm'));
B04_6 = double(imread('yaleB04_P00A-020E-10.pgm'));
B04_7 = double(imread('yaleB04_P00A-020E-40.pgm'));
B04_8 = double(imread('yaleB04_P00A-020E+10.pgm'));
B04_9 = double(imread('yaleB04_P00A-025E+00.pgm'));
B04_10 = double(imread('yaleB04_P00A-035E-20.pgm'));
B04_11 = double(imread('yaleB04_P00A-035E+15.pgm'));
B04_12 = double(imread('yaleB04_P00A-035E+40.pgm'));
B04_13 = double(imread('yaleB04_P00A-035E+65.pgm'));
B04_14 = double(imread('yaleB04_P00A-050E-40.pgm'));
B04_15 = double(imread('yaleB04_P00A-050E+00.pgm'));
B04_16 = double(imread('yaleB04_P00A-060E-20.pgm'));
B04_17 = double(imread('yaleB04_P00A-060E+20.pgm'));
B04_18 = double(imread('yaleB04_P00A-070E-35.pgm'));
B04_19 = double(imread('yaleB04_P00A-070E+00.pgm'));
B04_20 = double(imread('yaleB04_P00A-070E+45.pgm'));
B04_21 = double(imread('yaleB04_P00A-085E-20.pgm'));
B04_22 = double(imread('yaleB04_P00A-085E+20.pgm'));
B04_23 = double(imread('yaleB04_P00A-095E+00.pgm'));
B04_24 = double(imread('yaleB04_P00A-110E-20.pgm'));
B04_25 = double(imread('yaleB04_P00A-110E+15.pgm'));
B04_26 = double(imread('yaleB04_P00A-110E+40.pgm'));
B04_27 = double(imread('yaleB04_P00A-110E+65.pgm'));
B04_28 = double(imread('yaleB04_P00A-120E+00.pgm'));
B04_29 = double(imread('yaleB04_P00A-130E+20.pgm'));
B04_30 = double(imread('yaleB04_P00A+000E-20.pgm'));
B04_31 = double(imread('yaleB04_P00A+000E-35.pgm'));
B04_32 = double(imread('yaleB04_P00A+000E+00.pgm'));
B04_33 = double(imread('yaleB04_P00A+000E+20.pgm'));
B04_34 = double(imread('yaleB04_P00A+000E+45.pgm'));
B04_35 = double(imread('yaleB04_P00A+000E+90.pgm'));
B04_36 = double(imread('yaleB04_P00A+005E-10.pgm'));
B04_37 = double(imread('yaleB04_P00A+005E+10.pgm'));
B04_38 = double(imread('yaleB04_P00A+010E-20.pgm'));
B04_39 = double(imread('yaleB04_P00A+010E+00.pgm'));
B04_40 = double(imread('yaleB04_P00A+015E+20.pgm'));
B04_41 = double(imread('yaleB04_P00A+020E-10.pgm'));
B04_42 = double(imread('yaleB04_P00A+020E-40.pgm'));
B04_43 = double(imread('yaleB04_P00A+020E+10.pgm'));
B04_44 = double(imread('yaleB04_P00A+025E+00.pgm'));
B04_45 = double(imread('yaleB04_P00A+035E-20.pgm'));
B04_46 = double(imread('yaleB04_P00A+035E+15.pgm'));
B04_47 = double(imread('yaleB04_P00A+035E+40.pgm'));
B04_48 = double(imread('yaleB04_P00A+035E+65.pgm'));
B04_49 = double(imread('yaleB04_P00A+050E-40.pgm'));
B04_50 = double(imread('yaleB04_P00A+050E+00.pgm'));
B04_51 = double(imread('yaleB04_P00A+060E-20.pgm'));
B04_52 = double(imread('yaleB04_P00A+060E+20.pgm'));
B04_53 = double(imread('yaleB04_P00A+070E-35.pgm'));
B04_54 = double(imread('yaleB04_P00A+070E+00.pgm'));
B04_55 = double(imread('yaleB04_P00A+070E+45.pgm'));
B04_56 = double(imread('yaleB04_P00A+085E-20.pgm'));
B04_57 = double(imread('yaleB04_P00A+085E+20.pgm'));
B04_58 = double(imread('yaleB04_P00A+095E+00.pgm'));
B04_59 = double(imread('yaleB04_P00A+110E-20.pgm'));
B04_60 = double(imread('yaleB04_P00A+110E+15.pgm'));
B04_61 = double(imread('yaleB04_P00A+110E+40.pgm'));
B04_62 = double(imread('yaleB04_P00A+110E+65.pgm'));
B04_63 = double(imread('yaleB04_P00A+120E+00.pgm'));
```

```
824 B04_64 = double(imread('yaleB04_P00A+130E+20.pgm'));
825
826 B04_Ave = (B04_1 + B04_2 + B04_3 + B04_4 + B04_5 + B04_6 + B04_7 + B04_8 + B04_9 + B04_10 ...
        + B04_11 + B04_12 + B04_13 + B04_14 + B04_15 + B04_16 + B04_17 + B04_18 + B04_19 + ...
        B04_20 + B04_21 + B04_22 + B04_23 + B04_24 + B04_25 + B04_26 + B04_27 + B04_28 + ...
        B04_29 +  B04_30 + B04_31 + B04_32 + B04_33 + B04_34 + B04_35 + B04_36 + B04_37 + ...
        B04_38 + B04_39 + B04_40 + B04_41 + B04_42 + B04_43 + B04_44 + B04_45 + B04_46 + ...
        B04_47 + B04_48 + B04_49+ B04_50 + B04_51 + B04_52 + B04_53 + B04_54 + B04_55 + B04_56 ...
        + B04_57 + B04_58 + B04_59 + B04_60 + B04_61 + B04_62 + B04_63 + B04_64)/64;
827
828 % Subject 5
829 B05_1 = double(imread('yaleB05_P00A-005E-10.pgm'));
830 B05_2 = double(imread('yaleB05_P00A-005E+10.pgm'));
831 B05_3 = double(imread('yaleB05_P00A-010E-20.pgm'));
832 B05_4 = double(imread('yaleB05_P00A-010E+00.pgm'));
833 B05_5 = double(imread('yaleB05_P00A-015E+20.pgm'));
834 B05_6 = double(imread('yaleB05_P00A-020E-10.pgm'));
835 B05_7 = double(imread('yaleB05_P00A-020E-40.pgm'));
836 B05_8 = double(imread('yaleB05_P00A-020E+10.pgm'));
837 B05_9 = double(imread('yaleB05_P00A-025E+00.pgm'));
838 B05_10 = double(imread('yaleB05_P00A-035E-20.pgm'));
839 B05_11 = double(imread('yaleB05_P00A-035E+15.pgm'));
840 B05_12 = double(imread('yaleB05_P00A-035E+40.pgm'));
841 B05_13 = double(imread('yaleB05_P00A-035E+65.pgm'));
842 B05_14 = double(imread('yaleB05_P00A-050E-40.pgm'));
843 B05_15 = double(imread('yaleB05_P00A-050E+00.pgm'));
844 B05_16 = double(imread('yaleB05_P00A-060E-20.pgm'));
845 B05_17 = double(imread('yaleB05_P00A-060E+20.pgm'));
846 B05_18 = double(imread('yaleB05_P00A-070E-35.pgm'));
847 B05_19 = double(imread('yaleB05_P00A-070E+00.pgm'));
848 B05_20 = double(imread('yaleB05_P00A-070E+45.pgm'));
849 B05_21 = double(imread('yaleB05_P00A-085E-20.pgm'));
850 B05_22 = double(imread('yaleB05_P00A-085E+20.pgm'));
851 B05_23 = double(imread('yaleB05_P00A-095E+00.pgm'));
852 B05_24 = double(imread('yaleB05_P00A-110E-20.pgm'));
853 B05_25 = double(imread('yaleB05_P00A-110E+15.pgm'));
854 B05_26 = double(imread('yaleB05_P00A-110E+40.pgm'));
855 B05_27 = double(imread('yaleB05_P00A-110E+65.pgm'));
856 B05_28 = double(imread('yaleB05_P00A-120E+00.pgm'));
857 B05_29 = double(imread('yaleB05_P00A-130E+20.pgm'));
858 B05_30 = double(imread('yaleB05_P00A+000E-20.pgm'));
859 B05_31 = double(imread('yaleB05_P00A+000E-35.pgm'));
860 B05_32 = double(imread('yaleB05_P00A+000E+00.pgm'));
861 B05_33 = double(imread('yaleB05_P00A+000E+20.pgm'));
862 B05_34 = double(imread('yaleB05_P00A+000E+45.pgm'));
863 B05_35 = double(imread('yaleB05_P00A+000E+90.pgm'));
864 B05_36 = double(imread('yaleB05_P00A+005E-10.pgm'));
865 B05_37 = double(imread('yaleB05_P00A+005E+10.pgm'));
866 B05_38 = double(imread('yaleB05_P00A+010E-20.pgm'));
867 B05_39 = double(imread('yaleB05_P00A+010E+00.pgm'));
868 B05_40 = double(imread('yaleB05_P00A+015E+20.pgm'));
869 B05_41 = double(imread('yaleB05_P00A+020E-10.pgm'));
870 B05_42 = double(imread('yaleB05_P00A+020E-40.pgm'));
871 B05_43 = double(imread('yaleB05_P00A+020E+10.pgm'));
872 B05_44 = double(imread('yaleB05_P00A+025E+00.pgm'));
873 B05_45 = double(imread('yaleB05_P00A+035E-20.pgm'));
874 B05_46 = double(imread('yaleB05_P00A+035E+15.pgm'));
875 B05_47 = double(imread('yaleB05_P00A+035E+40.pgm'));
876 B05_48 = double(imread('yaleB05_P00A+035E+65.pgm'));
877 B05_49 = double(imread('yaleB05_P00A+050E-40.pgm'));
878 B05_50 = double(imread('yaleB05_P00A+050E+00.pgm'));
879 B05_51 = double(imread('yaleB05_P00A+060E-20.pgm'));
880 B05_52 = double(imread('yaleB05_P00A+060E+20.pgm'));
881 B05_53 = double(imread('yaleB05_P00A+070E-35.pgm'));
882 B05_54 = double(imread('yaleB05_P00A+070E+00.pgm'));
883 B05_55 = double(imread('yaleB05_P00A+070E+45.pgm'));
884 B05_56 = double(imread('yaleB05_P00A+085E-20.pgm'));
885 B05_57 = double(imread('yaleB05_P00A+085E+20.pgm'));
```

```
B05_58 = double(imread('yaleB05_P00A+095E+00.pgm'));
B05_59 = double(imread('yaleB05_P00A+110E-20.pgm'));
B05_60 = double(imread('yaleB05_P00A+110E+15.pgm'));
B05_61 = double(imread('yaleB05_P00A+110E+40.pgm'));
B05_62 = double(imread('yaleB05_P00A+110E+65.pgm'));
B05_63 = double(imread('yaleB05_P00A+120E+00.pgm'));
B05_64 = double(imread('yaleB05_P00A+130E+20.pgm'));

B05_Ave = (B05_1 + B05_2 + B05_3 + B05_4 + B05_5 + B05_6 + B05_7 + B05_8 + B05_9 + B05_10 ...
    + B05_11 + B05_12 + B05_13 + B05_14 + B05_15 + B05_16 + B05_17 + B05_18 + B05_19 + ...
    B05_20 + B05_21 + B05_22 + B05_23 + B05_24 + B05_25 + B05_26 + B05_27 + B05_28 + ...
    B05_29 +  B05_30 + B05_31 + B05_32 + B05_33 + B05_34 + B05_35 + B05_36 + B05_37 + ...
    B05_38 + B05_39 + B05_40 + B05_41 + B05_42 + B05_43 + B05_44 + B05_45 + B05_46 + ...
    B05_47 + B05_48 + B05_49+ B05_50 + B05_51 + B05_52 + B05_53 + B05_54 + B05_55 + B05_56 ...
    + B05_57 + B05_58 + B05_59 + B05_60 + B05_61 + B05_62 + B05_63 + B05_64)/64;

% Subject 6
B06_1 = double(imread('yaleB06_P00A-005E-10.pgm'));
B06_2 = double(imread('yaleB06_P00A-005E+10.pgm'));
B06_3 = double(imread('yaleB06_P00A-010E-20.pgm'));
B06_4 = double(imread('yaleB06_P00A-010E+00.pgm'));
B06_5 = double(imread('yaleB06_P00A-015E+20.pgm'));
B06_6 = double(imread('yaleB06_P00A-020E-10.pgm'));
B06_7 = double(imread('yaleB06_P00A-020E-40.pgm'));
B06_8 = double(imread('yaleB06_P00A-020E+10.pgm'));
B06_9 = double(imread('yaleB06_P00A-025E+00.pgm'));
B06_10 = double(imread('yaleB06_P00A-035E-20.pgm'));
B06_11 = double(imread('yaleB06_P00A-035E+15.pgm'));
B06_12 = double(imread('yaleB06_P00A-035E+40.pgm'));
B06_13 = double(imread('yaleB06_P00A-035E+65.pgm'));
B06_14 = double(imread('yaleB06_P00A-050E-40.pgm'));
B06_15 = double(imread('yaleB06_P00A-050E+00.pgm'));
B06_16 = double(imread('yaleB06_P00A-060E-20.pgm'));
B06_17 = double(imread('yaleB06_P00A-060E+20.pgm'));
B06_18 = double(imread('yaleB06_P00A-070E-35.pgm'));
B06_19 = double(imread('yaleB06_P00A-070E+00.pgm'));
B06_20 = double(imread('yaleB06_P00A-070E+45.pgm'));
B06_21 = double(imread('yaleB06_P00A-085E-20.pgm'));
B06_22 = double(imread('yaleB06_P00A-085E+20.pgm'));
B06_23 = double(imread('yaleB06_P00A-095E+00.pgm'));
B06_24 = double(imread('yaleB06_P00A-110E-20.pgm'));
B06_25 = double(imread('yaleB06_P00A-110E+15.pgm'));
B06_26 = double(imread('yaleB06_P00A-110E+40.pgm'));
B06_27 = double(imread('yaleB06_P00A-110E+65.pgm'));
B06_28 = double(imread('yaleB06_P00A-120E+00.pgm'));
B06_29 = double(imread('yaleB06_P00A-130E+20.pgm'));
B06_30 = double(imread('yaleB06_P00A+000E-20.pgm'));
B06_31 = double(imread('yaleB06_P00A+000E-35.pgm'));
B06_32 = double(imread('yaleB06_P00A+000E+00.pgm'));
B06_33 = double(imread('yaleB06_P00A+000E+20.pgm'));
B06_34 = double(imread('yaleB06_P00A+000E+45.pgm'));
B06_35 = double(imread('yaleB06_P00A+000E+90.pgm'));
B06_36 = double(imread('yaleB06_P00A+005E-10.pgm'));
B06_37 = double(imread('yaleB06_P00A+005E+10.pgm'));
B06_38 = double(imread('yaleB06_P00A+010E-20.pgm'));
B06_39 = double(imread('yaleB06_P00A+010E+00.pgm'));
B06_40 = double(imread('yaleB06_P00A+015E+20.pgm'));
B06_41 = double(imread('yaleB06_P00A+020E-10.pgm'));
B06_42 = double(imread('yaleB06_P00A+020E-40.pgm'));
B06_43 = double(imread('yaleB06_P00A+020E+10.pgm'));
B06_44 = double(imread('yaleB06_P00A+025E+00.pgm'));
B06_45 = double(imread('yaleB06_P00A+035E-20.pgm'));
B06_46 = double(imread('yaleB06_P00A+035E+15.pgm'));
B06_47 = double(imread('yaleB06_P00A+035E+40.pgm'));
B06_48 = double(imread('yaleB06_P00A+035E+65.pgm'));
B06_49 = double(imread('yaleB06_P00A+050E-40.pgm'));
B06_50 = double(imread('yaleB06_P00A+050E+00.pgm'));
B06_51 = double(imread('yaleB06_P00A+060E-20.pgm'));
```

```
B06_52 = double(imread('yaleB06_P00A+060E+20.pgm'));
B06_53 = double(imread('yaleB06_P00A+070E-35.pgm'));
B06_54 = double(imread('yaleB06_P00A+070E+00.pgm'));
B06_55 = double(imread('yaleB06_P00A+070E+45.pgm'));
B06_56 = double(imread('yaleB06_P00A+085E-20.pgm'));
B06_57 = double(imread('yaleB06_P00A+085E+20.pgm'));
B06_58 = double(imread('yaleB06_P00A+095E+00.pgm'));
B06_59 = double(imread('yaleB06_P00A+110E-20.pgm'));
B06_60 = double(imread('yaleB06_P00A+110E+15.pgm'));
B06_61 = double(imread('yaleB06_P00A+110E+40.pgm'));
B06_62 = double(imread('yaleB06_P00A+110E+65.pgm'));
B06_63 = double(imread('yaleB06_P00A+120E+00.pgm'));
B06_64 = double(imread('yaleB06_P00A+130E+20.pgm'));

B06_Ave = (B06_1 + B06_2 + B06_3 + B06_4 + B06_5 + B06_6 + B06_7 + B06_8 + B06_9 + B06_10 ...
    + B06_11 + B06_12 + B06_13 + B06_14 + B06_15 + B06_16 + B06_17 + B06_18 + B06_19 + ...
    B06_20 + B06_21 + B06_22 + B06_23 + B06_24 + B06_25 + B06_26 + B06_27 + B06_28 + ...
    B06_29 +  B06_30 + B06_31 + B06_32 + B06_33 + B06_34 + B06_35 + B06_36 + B06_37 + ...
    B06_38 + B06_39 + B06_40 + B06_41 + B06_42 + B06_43 + B06_44 + B06_45 + B06_46 + ...
    B06_47 + B06_48 + B06_49+ B06_50 + B06_51 + B06_52 + B06_53 + B06_54 + B06_55 + B06_56 ...
    + B06_57 + B06_58 + B06_59 + B06_60 + B06_61 + B06_62 + B06_63 + B06_64)/64;

% Subject 7
B07_1 = double(imread('yaleB07_P00A-005E-10.pgm'));
B07_2 = double(imread('yaleB07_P00A-005E+10.pgm'));
B07_3 = double(imread('yaleB07_P00A-010E-20.pgm'));
B07_4 = double(imread('yaleB07_P00A-010E+00.pgm'));
B07_5 = double(imread('yaleB07_P00A-015E+20.pgm'));
B07_6 = double(imread('yaleB07_P00A-020E-10.pgm'));
B07_7 = double(imread('yaleB07_P00A-020E-40.pgm'));
B07_8 = double(imread('yaleB07_P00A-020E+10.pgm'));
B07_9 = double(imread('yaleB07_P00A-025E+00.pgm'));
B07_10 = double(imread('yaleB07_P00A-035E-20.pgm'));
B07_11 = double(imread('yaleB07_P00A-035E+15.pgm'));
B07_12 = double(imread('yaleB07_P00A-035E+40.pgm'));
B07_13 = double(imread('yaleB07_P00A-035E+65.pgm'));
B07_14 = double(imread('yaleB07_P00A-050E-40.pgm'));
B07_15 = double(imread('yaleB07_P00A-050E+00.pgm'));
B07_16 = double(imread('yaleB07_P00A-060E-20.pgm'));
B07_17 = double(imread('yaleB07_P00A-060E+20.pgm'));
B07_18 = double(imread('yaleB07_P00A-070E-35.pgm'));
B07_19 = double(imread('yaleB07_P00A-070E+00.pgm'));
B07_20 = double(imread('yaleB07_P00A-070E+45.pgm'));
B07_21 = double(imread('yaleB07_P00A-085E-20.pgm'));
B07_22 = double(imread('yaleB07_P00A-085E+20.pgm'));
B07_23 = double(imread('yaleB07_P00A-095E+00.pgm'));
B07_24 = double(imread('yaleB07_P00A-110E-20.pgm'));
B07_25 = double(imread('yaleB07_P00A-110E+15.pgm'));
B07_26 = double(imread('yaleB07_P00A-110E+40.pgm'));
B07_27 = double(imread('yaleB07_P00A-110E+65.pgm'));
B07_28 = double(imread('yaleB07_P00A-120E+00.pgm'));
B07_29 = double(imread('yaleB07_P00A-130E+20.pgm'));
B07_30 = double(imread('yaleB07_P00A+000E-20.pgm'));
B07_31 = double(imread('yaleB07_P00A+000E-35.pgm'));
B07_32 = double(imread('yaleB07_P00A+000E+00.pgm'));
B07_33 = double(imread('yaleB07_P00A+000E+20.pgm'));
B07_34 = double(imread('yaleB07_P00A+000E+45.pgm'));
B07_35 = double(imread('yaleB07_P00A+000E+90.pgm'));
B07_36 = double(imread('yaleB07_P00A+005E-10.pgm'));
B07_37 = double(imread('yaleB07_P00A+005E+10.pgm'));
B07_38 = double(imread('yaleB07_P00A+010E-20.pgm'));
B07_39 = double(imread('yaleB07_P00A+010E+00.pgm'));
B07_40 = double(imread('yaleB07_P00A+015E+20.pgm'));
B07_41 = double(imread('yaleB07_P00A+020E-10.pgm'));
B07_42 = double(imread('yaleB07_P00A+020E-40.pgm'));
B07_43 = double(imread('yaleB07_P00A+020E+10.pgm'));
B07_44 = double(imread('yaleB07_P00A+025E+00.pgm'));
B07_45 = double(imread('yaleB07_P00A+035E-20.pgm'));
```

```
B07_46 = double(imread('yaleB07_P00A+035E+15.pgm'));
B07_47 = double(imread('yaleB07_P00A+035E+40.pgm'));
B07_48 = double(imread('yaleB07_P00A+035E+65.pgm'));
B07_49 = double(imread('yaleB07_P00A+050E-40.pgm'));
B07_50 = double(imread('yaleB07_P00A+050E+00.pgm'));
B07_51 = double(imread('yaleB07_P00A+060E-20.pgm'));
B07_52 = double(imread('yaleB07_P00A+060E+20.pgm'));
B07_53 = double(imread('yaleB07_P00A+070E-35.pgm'));
B07_54 = double(imread('yaleB07_P00A+070E+00.pgm'));
B07_55 = double(imread('yaleB07_P00A+070E+45.pgm'));
B07_56 = double(imread('yaleB07_P00A+085E-20.pgm'));
B07_57 = double(imread('yaleB07_P00A+085E+20.pgm'));
B07_58 = double(imread('yaleB07_P00A+095E+00.pgm'));
B07_59 = double(imread('yaleB07_P00A+110E-20.pgm'));
B07_60 = double(imread('yaleB07_P00A+110E+15.pgm'));
B07_61 = double(imread('yaleB07_P00A+110E+40.pgm'));
B07_62 = double(imread('yaleB07_P00A+110E+65.pgm'));
B07_63 = double(imread('yaleB07_P00A+120E+00.pgm'));
B07_64 = double(imread('yaleB07_P00A+130E+20.pgm'));

B07_Ave = (B07_1 + B07_2 + B07_3 + B07_4 + B07_5 + B07_6 + B07_7 + B07_8 + B07_9 + B07_10 ...
    + B07_11 + B07_12 + B07_13 + B07_14 + B07_15 + B07_16 + B07_17 + B07_18 + B07_19 + ...
    B07_20 + B07_21 + B07_22 + B07_23 + B07_24 + B07_25 + B07_26 + B07_27 + B07_28 + ...
    B07_29 +  B07_30 + B07_31 + B07_32 + B07_33 + B07_34 + B07_35 + B07_36 + B07_37 + ...
    B07_38 + B07_39 + B07_40 + B07_41 + B07_42 + B07_43 + B07_44 + B07_45 + B07_46 + ...
    B07_47 + B07_48 + B07_49+ B07_50 + B07_51 + B07_52 + B07_53 + B07_54 + B07_55 + B07_56 ...
    + B07_57 + B07_58 + B07_59 + B07_60 + B07_61 + B07_62 + B07_63 + B07_64)/64;

% Subject 8
B08_1 = double(imread('yaleB08_P00A-005E-10.pgm'));
B08_2 = double(imread('yaleB08_P00A-005E+10.pgm'));
B08_3 = double(imread('yaleB08_P00A-010E-20.pgm'));
B08_4 = double(imread('yaleB08_P00A-010E+00.pgm'));
B08_5 = double(imread('yaleB08_P00A-015E+20.pgm'));
B08_6 = double(imread('yaleB08_P00A-020E-10.pgm'));
B08_7 = double(imread('yaleB08_P00A-020E-40.pgm'));
B08_8 = double(imread('yaleB08_P00A-020E+10.pgm'));
B08_9 = double(imread('yaleB08_P00A-025E+00.pgm'));
B08_10 = double(imread('yaleB08_P00A-035E-20.pgm'));
B08_11 = double(imread('yaleB08_P00A-035E+15.pgm'));
B08_12 = double(imread('yaleB08_P00A-035E+40.pgm'));
B08_13 = double(imread('yaleB08_P00A-035E+65.pgm'));
B08_14 = double(imread('yaleB08_P00A-050E-40.pgm'));
B08_15 = double(imread('yaleB08_P00A-050E+00.pgm'));
B08_16 = double(imread('yaleB08_P00A-060E-20.pgm'));
B08_17 = double(imread('yaleB08_P00A-060E+20.pgm'));
B08_18 = double(imread('yaleB08_P00A-070E-35.pgm'));
B08_19 = double(imread('yaleB08_P00A-070E+00.pgm'));
B08_20 = double(imread('yaleB08_P00A-070E+45.pgm'));
B08_21 = double(imread('yaleB08_P00A-085E-20.pgm'));
B08_22 = double(imread('yaleB08_P00A-085E+20.pgm'));
B08_23 = double(imread('yaleB08_P00A-095E+00.pgm'));
B08_24 = double(imread('yaleB08_P00A-110E-20.pgm'));
B08_25 = double(imread('yaleB08_P00A-110E+15.pgm'));
B08_26 = double(imread('yaleB08_P00A-110E+40.pgm'));
B08_27 = double(imread('yaleB08_P00A-110E+65.pgm'));
B08_28 = double(imread('yaleB08_P00A-120E+00.pgm'));
B08_29 = double(imread('yaleB08_P00A-130E+20.pgm'));
B08_30 = double(imread('yaleB08_P00A+000E-20.pgm'));
B08_31 = double(imread('yaleB08_P00A+000E-35.pgm'));
B08_32 = double(imread('yaleB08_P00A+000E+00.pgm'));
B08_33 = double(imread('yaleB08_P00A+000E+20.pgm'));
B08_34 = double(imread('yaleB08_P00A+000E+45.pgm'));
B08_35 = double(imread('yaleB08_P00A+000E+90.pgm'));
B08_36 = double(imread('yaleB08_P00A+005E-10.pgm'));
B08_37 = double(imread('yaleB08_P00A+005E+10.pgm'));
B08_38 = double(imread('yaleB08_P00A+010E-20.pgm'));
B08_39 = double(imread('yaleB08_P00A+010E+00.pgm'));
```

```
B08_40 = double(imread('yaleB08_P00A+015E+20.pgm'));
B08_41 = double(imread('yaleB08_P00A+020E-10.pgm'));
B08_42 = double(imread('yaleB08_P00A+020E-40.pgm'));
B08_43 = double(imread('yaleB08_P00A+020E+10.pgm'));
B08_44 = double(imread('yaleB08_P00A+025E+00.pgm'));
B08_45 = double(imread('yaleB08_P00A+035E-20.pgm'));
B08_46 = double(imread('yaleB08_P00A+035E+15.pgm'));
B08_47 = double(imread('yaleB08_P00A+035E+40.pgm'));
B08_48 = double(imread('yaleB08_P00A+035E+65.pgm'));
B08_49 = double(imread('yaleB08_P00A+050E-40.pgm'));
B08_50 = double(imread('yaleB08_P00A+050E+00.pgm'));
B08_51 = double(imread('yaleB08_P00A+060E-20.pgm'));
B08_52 = double(imread('yaleB08_P00A+060E+20.pgm'));
B08_53 = double(imread('yaleB08_P00A+070E-35.pgm'));
B08_54 = double(imread('yaleB08_P00A+070E+00.pgm'));
B08_55 = double(imread('yaleB08_P00A+070E+45.pgm'));
B08_56 = double(imread('yaleB08_P00A+085E-20.pgm'));
B08_57 = double(imread('yaleB08_P00A+085E+20.pgm'));
B08_58 = double(imread('yaleB08_P00A+095E+00.pgm'));
B08_59 = double(imread('yaleB08_P00A+110E-20.pgm'));
B08_60 = double(imread('yaleB08_P00A+110E+15.pgm'));
B08_61 = double(imread('yaleB08_P00A+110E+40.pgm'));
B08_62 = double(imread('yaleB08_P00A+110E+65.pgm'));
B08_63 = double(imread('yaleB08_P00A+120E+00.pgm'));
B08_64 = double(imread('yaleB08_P00A+130E+20.pgm'));

B08_Ave = (B08_1 + B08_2 + B08_3 + B08_4 + B08_5 + B08_6 + B08_7 + B08_8 + B08_9 + B08_10 ...
    + B08_11 + B08_12 + B08_13 + B08_14 + B08_15 + B08_16 + B08_17 + B08_18 + B08_19 + ...
    B08_20 + B08_21 + B08_22 + B08_23 + B08_24 + B08_25 + B08_26 + B08_27 + B08_28 + ...
    B08_29 +  B08_30 + B08_31 + B08_32 + B08_33 + B08_34 + B08_35 + B08_36 + B08_37 + ...
    B08_38 + B08_39 + B08_40 + B08_41 + B08_42 + B08_43 + B08_44 + B08_45 + B08_46 + ...
    B08_47 + B08_48 + B08_49+ B08_50 + B08_51 + B08_52 + B08_53 + B08_54 + B08_55 + B08_56 ...
    + B08_57 + B08_58 + B08_59 + B08_60 + B08_61 + B08_62 + B08_63 + B08_64)/64;

% Subject 9
B09_1 = double(imread('yaleB09_P00A-005E-10.pgm'));
B09_2 = double(imread('yaleB09_P00A-005E+10.pgm'));
B09_3 = double(imread('yaleB09_P00A-010E-20.pgm'));
B09_4 = double(imread('yaleB09_P00A-010E+00.pgm'));
B09_5 = double(imread('yaleB09_P00A-015E+20.pgm'));
B09_6 = double(imread('yaleB09_P00A-020E-10.pgm'));
B09_7 = double(imread('yaleB09_P00A-020E-40.pgm'));
B09_8 = double(imread('yaleB09_P00A-020E+10.pgm'));
B09_9 = double(imread('yaleB09_P00A-025E+00.pgm'));
B09_10 = double(imread('yaleB09_P00A-035E-20.pgm'));
B09_11 = double(imread('yaleB09_P00A-035E+15.pgm'));
B09_12 = double(imread('yaleB09_P00A-035E+40.pgm'));
B09_13 = double(imread('yaleB09_P00A-035E+65.pgm'));
B09_14 = double(imread('yaleB09_P00A-050E-40.pgm'));
B09_15 = double(imread('yaleB09_P00A-050E+00.pgm'));
B09_16 = double(imread('yaleB09_P00A-060E-20.pgm'));
B09_17 = double(imread('yaleB09_P00A-060E+20.pgm'));
B09_18 = double(imread('yaleB09_P00A-070E-35.pgm'));
B09_19 = double(imread('yaleB09_P00A-070E+00.pgm'));
B09_20 = double(imread('yaleB09_P00A-070E+45.pgm'));
B09_21 = double(imread('yaleB09_P00A-085E-20.pgm'));
B09_22 = double(imread('yaleB09_P00A-085E+20.pgm'));
B09_23 = double(imread('yaleB09_P00A-095E+00.pgm'));
B09_24 = double(imread('yaleB09_P00A-110E-20.pgm'));
B09_25 = double(imread('yaleB09_P00A-110E+15.pgm'));
B09_26 = double(imread('yaleB09_P00A-110E+40.pgm'));
B09_27 = double(imread('yaleB09_P00A-110E+65.pgm'));
B09_28 = double(imread('yaleB09_P00A-120E+00.pgm'));
B09_29 = double(imread('yaleB09_P00A-130E+20.pgm'));
B09_30 = double(imread('yaleB09_P00A+000E-20.pgm'));
B09_31 = double(imread('yaleB09_P00A+000E-35.pgm'));
B09_32 = double(imread('yaleB09_P00A+000E+00.pgm'));
B09_33 = double(imread('yaleB09_P00A+000E+20.pgm'));
```

```
B09_34 = double(imread('yaleB09_P00A+000E+45.pgm'));
B09_35 = double(imread('yaleB09_P00A+000E+90.pgm'));
B09_36 = double(imread('yaleB09_P00A+005E-10.pgm'));
B09_37 = double(imread('yaleB09_P00A+005E+10.pgm'));
B09_38 = double(imread('yaleB09_P00A+010E-20.pgm'));
B09_39 = double(imread('yaleB09_P00A+010E+00.pgm'));
B09_40 = double(imread('yaleB09_P00A+015E+20.pgm'));
B09_41 = double(imread('yaleB09_P00A+020E-10.pgm'));
B09_42 = double(imread('yaleB09_P00A+020E-40.pgm'));
B09_43 = double(imread('yaleB09_P00A+020E+10.pgm'));
B09_44 = double(imread('yaleB09_P00A+025E+00.pgm'));
B09_45 = double(imread('yaleB09_P00A+035E-20.pgm'));
B09_46 = double(imread('yaleB09_P00A+035E+15.pgm'));
B09_47 = double(imread('yaleB09_P00A+035E+40.pgm'));
B09_48 = double(imread('yaleB09_P00A+035E+65.pgm'));
B09_49 = double(imread('yaleB09_P00A+050E-40.pgm'));
B09_50 = double(imread('yaleB09_P00A+050E+00.pgm'));
B09_51 = double(imread('yaleB09_P00A+060E-20.pgm'));
B09_52 = double(imread('yaleB09_P00A+060E+20.pgm'));
B09_53 = double(imread('yaleB09_P00A+070E-35.pgm'));
B09_54 = double(imread('yaleB09_P00A+070E+00.pgm'));
B09_55 = double(imread('yaleB09_P00A+070E+45.pgm'));
B09_56 = double(imread('yaleB09_P00A+085E-20.pgm'));
B09_57 = double(imread('yaleB09_P00A+085E+20.pgm'));
B09_58 = double(imread('yaleB09_P00A+095E+00.pgm'));
B09_59 = double(imread('yaleB09_P00A+110E-20.pgm'));
B09_60 = double(imread('yaleB09_P00A+110E+15.pgm'));
B09_61 = double(imread('yaleB09_P00A+110E+40.pgm'));
B09_62 = double(imread('yaleB09_P00A+110E+65.pgm'));
B09_63 = double(imread('yaleB09_P00A+120E+00.pgm'));
B09_64 = double(imread('yaleB09_P00A+130E+20.pgm'));

B09_Ave = (B09_1 + B09_2 + B09_3 + B09_4 + B09_5 + B09_6 + B09_7 + B09_8 + B09_9 + B09_10 ...
    + B09_11 + B09_12 + B09_13 + B09_14 + B09_15 + B09_16 + B09_17 + B09_18 + B09_19 + ...
    B09_20 + B09_21 + B09_22 + B09_23 + B09_24 + B09_25 + B09_26 + B09_27 + B09_28 + ...
    B09_29 +  B09_30 + B09_31 + B09_32 + B09_33 + B09_34 + B09_35 + B09_36 + B09_37 + ...
    B09_38 + B09_39 + B09_40 + B09_41 + B09_42 + B09_43 + B09_44 + B09_45 + B09_46 + ...
    B09_47 + B09_48 + B09_49+ B09_50 + B09_51 + B09_52 + B09_53 + B09_54 + B09_55 + B09_56 ...
    + B09_57 + B09_58 + B09_59 + B09_60 + B09_61 + B09_62 + B09_63 + B09_64)/64;

% Subject 10
B10_1 = double(imread('yaleB10_P00A-005E-10.pgm'));
B10_2 = double(imread('yaleB10_P00A-005E+10.pgm'));
B10_3 = double(imread('yaleB10_P00A-010E-20.pgm'));
B10_4 = double(imread('yaleB10_P00A-010E+00.pgm'));
B10_5 = double(imread('yaleB10_P00A-015E+20.pgm'));
B10_6 = double(imread('yaleB10_P00A-020E-10.pgm'));
B10_7 = double(imread('yaleB10_P00A-020E-40.pgm'));
B10_8 = double(imread('yaleB10_P00A-020E+10.pgm'));
B10_9 = double(imread('yaleB10_P00A-025E+00.pgm'));
B10_10 = double(imread('yaleB10_P00A-035E-20.pgm'));
B10_11 = double(imread('yaleB10_P00A-035E+15.pgm'));
B10_12 = double(imread('yaleB10_P00A-035E+40.pgm'));
B10_13 = double(imread('yaleB10_P00A-035E+65.pgm'));
B10_14 = double(imread('yaleB10_P00A-050E-40.pgm'));
B10_15 = double(imread('yaleB10_P00A-050E+00.pgm'));
B10_16 = double(imread('yaleB10_P00A-060E-20.pgm'));
B10_17 = double(imread('yaleB10_P00A-060E+20.pgm'));
B10_18 = double(imread('yaleB10_P00A-070E-35.pgm'));
B10_19 = double(imread('yaleB10_P00A-070E+00.pgm'));
B10_20 = double(imread('yaleB10_P00A-070E+45.pgm'));
B10_21 = double(imread('yaleB10_P00A-085E-20.pgm'));
B10_22 = double(imread('yaleB10_P00A-085E+20.pgm'));
B10_23 = double(imread('yaleB10_P00A-095E+00.pgm'));
B10_24 = double(imread('yaleB10_P00A-110E-20.pgm'));
B10_25 = double(imread('yaleB10_P00A-110E+15.pgm'));
B10_26 = double(imread('yaleB10_P00A-110E+40.pgm'));
B10_27 = double(imread('yaleB10_P00A-110E+65.pgm'));
```

```
B10_28 = double(imread('yaleB10_P00A-120E+00.pgm'));
B10_29 = double(imread('yaleB10_P00A-130E+20.pgm'));
B10_30 = double(imread('yaleB10_P00A+000E-20.pgm'));
B10_31 = double(imread('yaleB10_P00A+000E-35.pgm'));
B10_32 = double(imread('yaleB10_P00A+000E+00.pgm'));
B10_33 = double(imread('yaleB10_P00A+000E+20.pgm'));
B10_34 = double(imread('yaleB10_P00A+000E+45.pgm'));
B10_35 = double(imread('yaleB10_P00A+000E+90.pgm'));
B10_36 = double(imread('yaleB10_P00A+005E-10.pgm'));
B10_37 = double(imread('yaleB10_P00A+005E+10.pgm'));
B10_38 = double(imread('yaleB10_P00A+010E-20.pgm'));
B10_39 = double(imread('yaleB10_P00A+010E+00.pgm'));
B10_40 = double(imread('yaleB10_P00A+015E+20.pgm'));
B10_41 = double(imread('yaleB10_P00A+020E-10.pgm'));
B10_42 = double(imread('yaleB10_P00A+020E-40.pgm'));
B10_43 = double(imread('yaleB10_P00A+020E+10.pgm'));
B10_44 = double(imread('yaleB10_P00A+025E+00.pgm'));
B10_45 = double(imread('yaleB10_P00A+035E-20.pgm'));
B10_46 = double(imread('yaleB10_P00A+035E+15.pgm'));
B10_47 = double(imread('yaleB10_P00A+035E+40.pgm'));
B10_48 = double(imread('yaleB10_P00A+035E+65.pgm'));
B10_49 = double(imread('yaleB10_P00A+050E-40.pgm'));
B10_50 = double(imread('yaleB10_P00A+050E+00.pgm'));
B10_51 = double(imread('yaleB10_P00A+060E-20.pgm'));
B10_52 = double(imread('yaleB10_P00A+060E+20.pgm'));
B10_53 = double(imread('yaleB10_P00A+070E-35.pgm'));
B10_54 = double(imread('yaleB10_P00A+070E+00.pgm'));
B10_55 = double(imread('yaleB10_P00A+070E+45.pgm'));
B10_56 = double(imread('yaleB10_P00A+085E-20.pgm'));
B10_57 = double(imread('yaleB10_P00A+085E+20.pgm'));
B10_58 = double(imread('yaleB10_P00A+095E+00.pgm'));
B10_59 = double(imread('yaleB10_P00A+110E-20.pgm'));
B10_60 = double(imread('yaleB10_P00A+110E+15.pgm'));
B10_61 = double(imread('yaleB10_P00A+110E+40.pgm'));
B10_62 = double(imread('yaleB10_P00A+110E+65.pgm'));
B10_63 = double(imread('yaleB10_P00A+120E+00.pgm'));
B10_64 = double(imread('yaleB10_P00A+130E+20.pgm'));

B10_Ave = (B10_1 + B10_2 + B10_3 + B10_4 + B10_5 + B10_6 + B10_7 + B10_8 + B10_9 + B10_10 ...
    + B10_11 + B10_12 + B10_13 + B10_14 + B10_15 + B10_16 + B10_17 + B10_18 + B10_19 + ...
    B10_20 + B10_21 + B10_22 + B10_23 + B10_24 + B10_25 + B10_26 + B10_27 + B10_28 + ...
    B10_29 +  B10_30 + B10_31 + B10_32 + B10_33 + B10_34 + B10_35 + B10_36 + B10_37 + ...
    B10_38 + B10_39 + B10_40 + B10_41 + B10_42 + B10_43 + B10_44 + B10_45 + B10_46 + ...
    B10_47 + B10_48 + B10_49+ B10_50 + B10_51 + B10_52 + B10_53 + B10_54 + B10_55 + B10_56 ...
    + B10_57 + B10_58 + B10_59 + B10_60 + B10_61 + B10_62 + B10_63 + B10_64)/64;



B_All_Ave = (B01_Ave + B02_Ave + B03_Ave + B04_Ave + B05_Ave + B06_Ave + B07_Ave + B08_Ave ...
    + B09_Ave + B10_Ave)/10;
%% Plot Average of Subjects
figure()
for j = 1:9
subplot (2,5,j)
 i = (['B0',num2str(j),'_Ave']);
 k = (['Subject ', num2str(j)]);
 pcolor(flipud(eval(i))), shading interp, colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
 title(k)
end
subplot (2,5,10)
 pcolor(flipud(B10_Ave)), shading interp, colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
 title('Subject 10')
%% Average of all Images
figure()
pcolor(flipud(B_All_Ave)), shading interp, colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
 title('Average of All Images')

 %% Create Data Matrix
```

```
for jj = 1:9
    for kk = 1:64
        i = (['B0', num2str(jj),'_',num2str(kk)]);
        D(kk+(jj-1)*64,:) = reshape(imresize(eval(i), [96, 84]), 1, 96*84);
    end
end
  for kk = 1:64
      i = (['B10','_',num2str(kk)]);
        D(kk+(9)*64,:) = reshape(imresize(eval(i), [96,84]), 1, 96*84);
  end
%% Eigenvalue Decomposition
A = (D')*D;
size(A)
[V,D] = eigs(A,20, 'lm');
%% svd
[u,s,v]=svd(A);
sig = diag(s);
figure()
semilogy(diag(sig)^2, 'ko', 'Linewidth', [2])
set (gca, 'Fontsize' , [14])
title('Diagonals')
%% Plot Diagonals
figure()
semilogy(diag(D), 'ko', 'Linewidth', [2])
set (gca, 'Fontsize' , [14])
title('Diagonals')
%% Plot Dominate 4
figure()

subplot(2,2,1), face1 = reshape(V(:,1),96, 84); pcolor(flipud(face1)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 1')
subplot(2,2,2), face2 = reshape(V(:,2),96, 84); pcolor(flipud(face2)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 2')
subplot(2,2,3), face3 = reshape(V(:,3),96, 84); pcolor(flipud(face3)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 3')
subplot(2,2,4), face4 = reshape(V(:,4),96, 84); pcolor(flipud(face4)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Face 4')
%% Reshape all Subject averages
vecB_1=reshape(imresize(B01_Ave, [96, 84]), 1, 96*84);
vecB_2=reshape(imresize(B02_Ave, [96, 84]), 1, 96*84);
vecB_3=reshape(imresize(B03_Ave, [96, 84]), 1, 96*84);
vecB_4=reshape(imresize(B04_Ave, [96, 84]), 1, 96*84);
vecB_5=reshape(imresize(B05_Ave, [96, 84]), 1, 96*84);
vecB_6=reshape(imresize(B06_Ave, [96, 84]), 1, 96*84);
vecB_7=reshape(imresize(B07_Ave, [96, 84]), 1, 96*84);
vecB_8=reshape(imresize(B08_Ave, [96, 84]), 1, 96*84);
vecB_9=reshape(imresize(B09_Ave, [96, 84]), 1, 96*84);
vecB_10=reshape(imresize(B10_Ave, [96, 84]), 1, 96*84);

%% Project Subject averages onto V
figure()
projB_1 = vecB_1*V;
subplot(5,2,1), bar(projB_1(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 1', 'Fontsize', [15])

projB_2 = vecB_2*V;
subplot(5,2,2), bar(projB_2(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 2', 'Fontsize', [15])

projB_3 = vecB_3*V;
subplot(5,2,3), bar(projB_3(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
```

```
text( 12, -1700, 'Subject 3', 'Fontsize', [15])

projB_4 = vecB_4*V;
subplot(5,2,4), bar(projB_4(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 4', 'Fontsize', [15])

projB_5 = vecB_5*V;
subplot(5,2,5), bar(projB_5(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 5', 'Fontsize', [15])

projB_6 = vecB_6*V;
subplot(5,2,6), bar(projB_6(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 6', 'Fontsize', [15])

projB_7 = vecB_7*V;
subplot(5,2,7), bar(projB_7(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 7', 'Fontsize', [15])

projB_8 = vecB_8*V;
subplot(5,2,8), bar(projB_8(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 8', 'Fontsize', [15])

projB_9 = vecB_9*V;
subplot(5,2,9), bar(projB_9(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 9', 'Fontsize', [15])

projB_10 = vecB_10*V;
subplot(5,2,10), bar(projB_10(2:20)), set(gca, 'Xlim', [0,20], 'Ylim', [-2000, 2000], ...
    'Xtick', [], 'Ytick',[])
text( 12, -1700, 'Subject 10', 'Fontsize', [15])


 %AMATH 582 Homework 4 Cropped
clear all; close all; clc; %Start Fresh

%% Load Rock
for k = 1:8
    i =(['Rock_', num2str(k), '.mp3']);
[y,Fs] = audioread(i);
tr_song = 30;  % record time in seconds
 Fs=length(y)/tr_song;
 plot((1:length(y))/Fs,y);
 xlabel('Time [sec]'); ylabel('Amplitude');
 title('Classical 1');  drawnow
 F = getframe(gcf);
[X, Map] = frame2im(F);
 X = rgb2gray(X);
 X = reshape(double(imresize(X, [105, 140])), 1, 105*140);
C(k+8,:) = X;
end

%% Load Rap Music
for k = 1:8
    i =(['Rap_', num2str(k), '.mp3']);
[y,Fs] = audioread(i);
tr_song = 30;  % record time in seconds
 Fs=length(y)/tr_song;
 plot((1:length(y))/Fs,y);
 xlabel('Time [sec]'); ylabel('Amplitude');
 title('Classical 1');  drawnow
 F = getframe(gcf);
[X, Map] = frame2im(F);
```

```
 X = rgb2gray(X);
 X = reshape(double(imresize(X, [105, 140])), 1, 105*140);
C(k+16,:) = X;
end
%% SVD
[u, s, v] = svd(C);
sig = diag(s);
%% Plot Diagonals
figure()
semilogy(sig, 'ko', 'Linewidth', [2])
set (gca, 'Xlim',[0 20], 'Fontsize' , [14])
title('Diagonals')

%% Plot Dominate 2
figure()
subplot(2,1,1), face1 = reshape(v(:,1),105, 140); pcolor(flipud(face1)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Sound 1')
subplot(2,1,2), face2 = reshape(v(:,2),105, 140); pcolor(flipud(face2)), shading interp, ...
    colormap(gray), set(gca, 'Xtick',[], 'Ytick',[])
title('Dominate Sound 2')
```