|  |
| --- |
| Face Detection |
|  |
| **Dhaval Malaviya** |
|  |

Contents

[HOW TO RUN MATLAB FUNCTION 2](#_Toc308387914)

[1. Eigenfaces 3](#_Toc308387915)

[i. Reading images from the training set and collecting in one matrix: 3](#_Toc308387916)

[ii. Averaging face of the training set. 4](#_Toc308387917)

[iii. Average Image Displayed 5](#_Toc308387918)

[b) PCA on the training set images 5](#_Toc308387919)

[c) Comparing the principle components 6](#_Toc308387920)

[d) N significant eigenvectors with largest associated eigenvalues 7](#_Toc308387921)

[e) Reading Images from test set, subtracting average image and projecting it onto eigenfaces 7](#_Toc308387922)

[Eigen Faces 8](#_Toc308387923)

[f) Reconstructing the test images 9](#_Toc308387924)

[g) Matching the test image with images in training image set 20](#_Toc308387925)

[2. Evaluation 26](#_Toc308387926)

[b) Recognition Rate 28](#_Toc308387927)

[c) Non Face Images 29](#_Toc308387928)

[d) Plot for Frobenius Norm Non Face Images 30](#_Toc308387929)

[e) Plot for Frobenius Norm for Face Images 31](#_Toc308387930)

# HOW TO RUN MATLAB FUNCTION

*faceDetection(numberOfTranImage,imageSeqDirectory,testImagePath,nonFaceImages,debug)*

1. numberOfTranImage = number of the images in training set. I have taken 54.
2. imageSeqDirectory = Directory path where training images are located. Please note only mention directory path and no need to specify the image name. My function will read all the images present under that folder.
3. testImagePath = Test Image path when nonFaceImages flag is set to 0 else, path of the folder where test images (face or non face) are stored.
4. debug = It is used to calculate the time taken for calculating eigenvalues and eigenvectors by PCA, XX’.

- debug = 1 (eigenvalues and eigenvectors for XX’ are calculated)

- debug = 2 (eigenvalues and eigenvectors calculated using PCA)

- debug = 3 (eigenvalues and eigenvectors of X’X are calculated)

Commands:

1. *faceDetection(54,”path\trainingSet','path\testSet\subject11.surprised',3)*

Above command is used for single test image.

1. *faceDetection(54,'path\trainingSet','path\NonfaceImages',1,3)*

Above command is used for multiple test images.

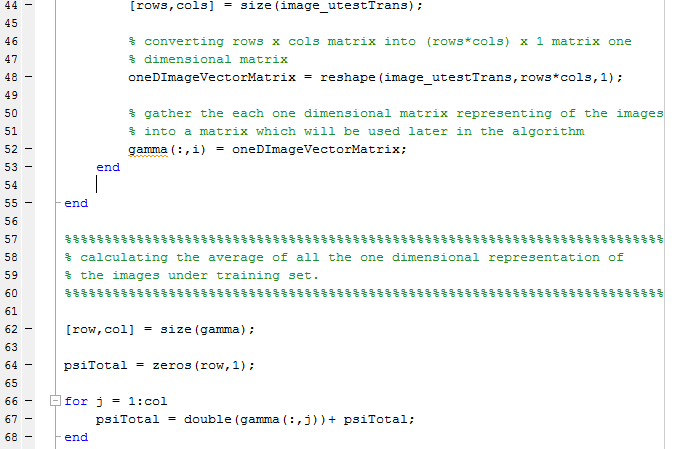
1. *faceDetection(54,'path\trainingSet','path\subject01.sad',2)*

Above command is used for calculating eigenvectors and eigenvalues using PCA.

1. Eigenfaces
2. Here in put to the function is path of the folder where training images are kept. Below piece of code will read all the image files in that particular folder and shall consider them to be training set images.
3. Reading images from the training set and collecting in one matrix:

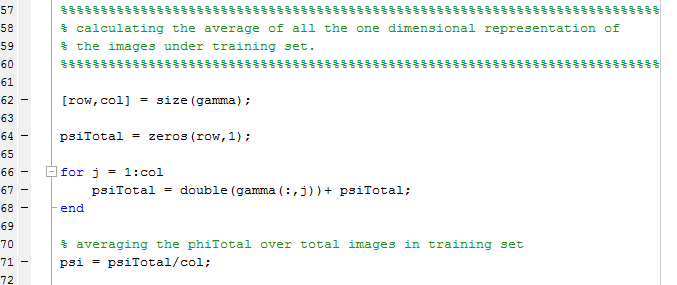
Matlab Code:





1. Averaging face of the training set.

Matlab Code:



1. Average Image Displayed

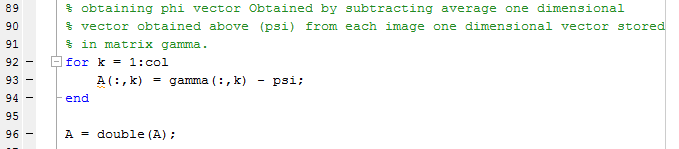


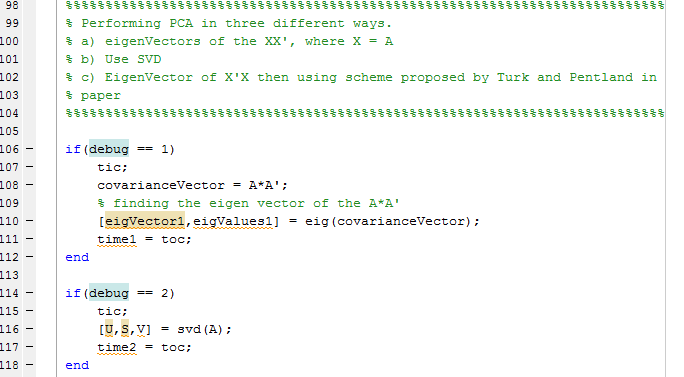
1. PCA on the training set images

Here in this part, user of the function needs to input ‘debug’ bit. If debug bit is 1 then, eigenvector of XX’ will be calculated and if debug bit is 2 then SVD will be calculated.

1. Eigenvectors of XX’ (In my case, its AA’)

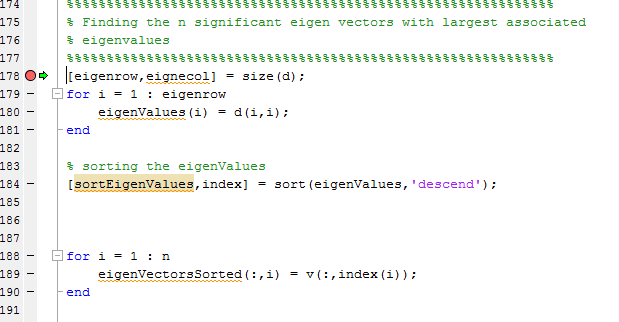
Matlab code:





1. Comparing the principle components
2. Time taken by SVD: 31.64
3. Time taken by X’X: 0.9529
4. N significant eigenvectors with largest associated eigenvalues

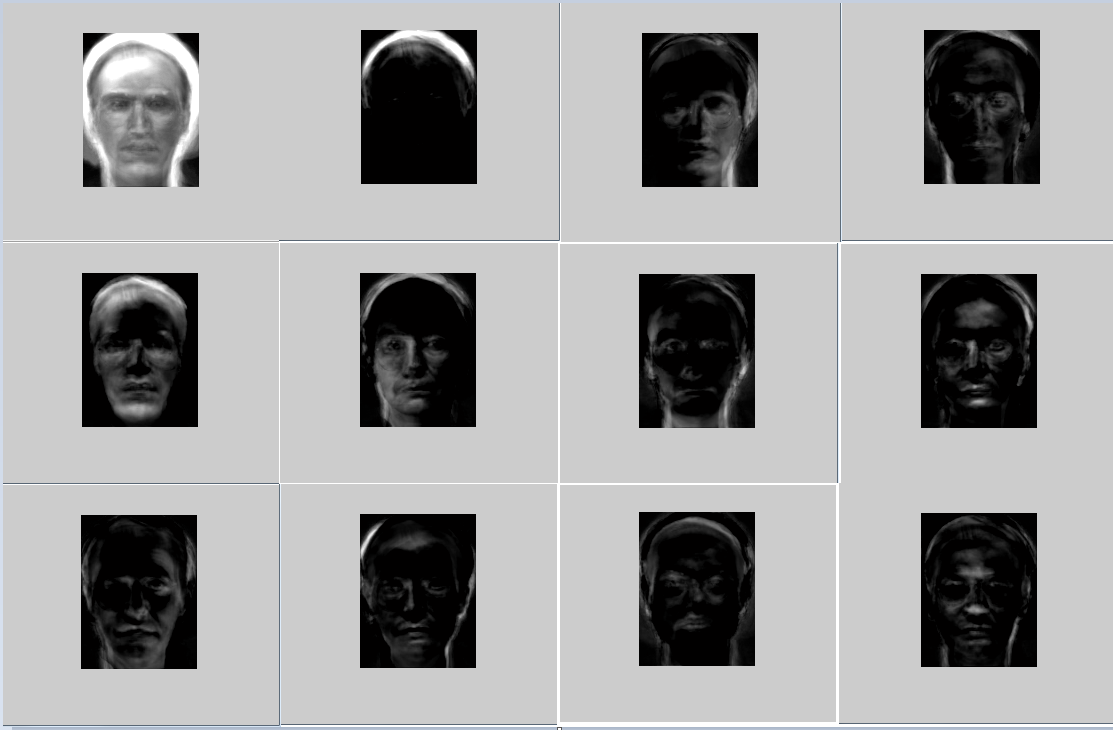
Matlab Code:



1. Reading Images from test set, subtracting average image and projecting it onto eigenfaces



Eigen Faces



1. Reconstructing the test images

Matlab Code:

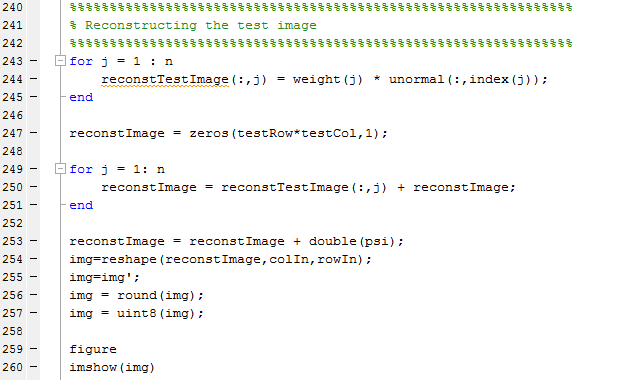


Image1 Original:



Projected Image:



Image 2 Original:



Projected Image:



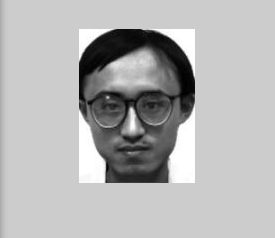
Image 3 Original:



Projected Image:



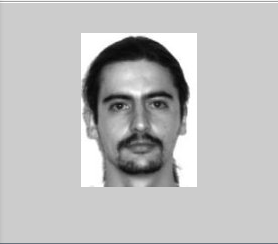
Image 4 Original:



Projected Image:



Image 5 Original:



Projected Image:

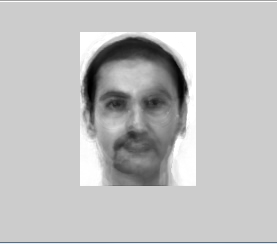
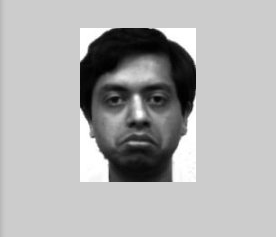


Image 6 Original:



Projected Image:



Image 7 Original:



Projected Image:



Image 8 Original:



Projected Image:



Image 9 Original:



Projected Image:



Image 10 Original:



Projected Image:



1. Matching the test image with images in training image set

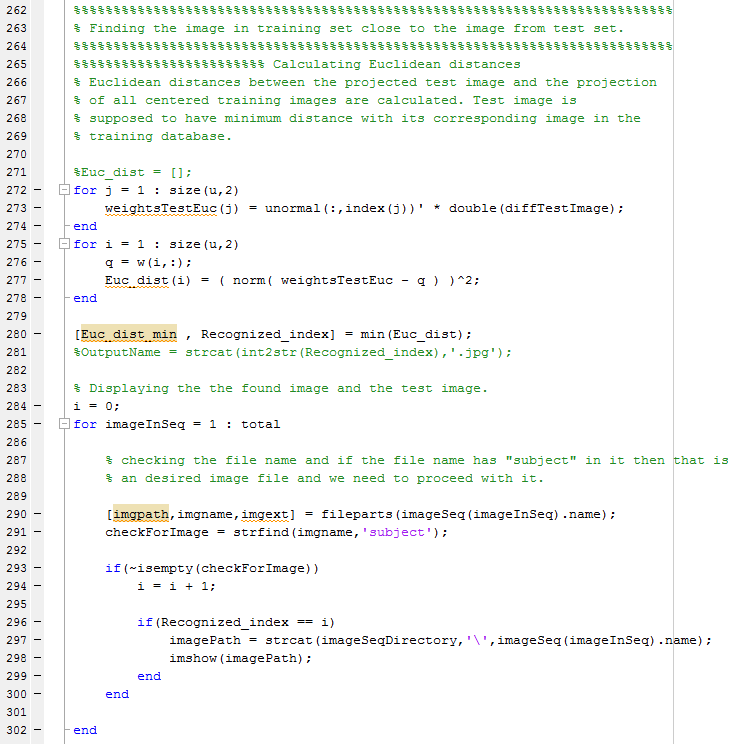


Image 1 Test Set Image:



Image 1 Closest Training Set Image:



Image 2 Test Set Image:



Image 2 Closest Training Set Image:



Image 3 Test Set Image:



Image 3 Closest Training Image:



Image 4 Test Set Image:



Image 4 Closest Training Set Image:



Image 5 Test Set Image:



Image 5 Closest Training Set Image:



Image 6 Test Set Image:



Test 6 Closest Training Set Image:



Image 7 Test Set Image:



Image 7 Closest Training Set Image:



Image 8 Test Set Image:



Image 8 Closest Training Set Image:



Image 9 Test Set Image:



Image 9 Closest Training Set Image:



Image 10 Test Set Image:



Image 10 Closest Training Set Image:



1. Evaluation
2. When n = 1,

Original Image:



Reconstructed Image:



When n = 2,



When n = 3



When n = 15



When n = 25



From above results, I found that after some n = 10, I am getting the reconstructed image which I can identify with some person.

However, for specific feature, for example in my case, person with glasses, good featured reconstructed image results only after n = 25.

1. Recognition Rate

Out of all 20 test images, I was able to find correct resulting (closest match of the same person) image in training set except one.

Hence recognition rate is 19/20.

1. Non Face Images

Projection on to the eigenfaces:

Image 1: Original Difference Projected

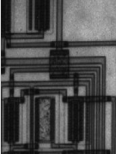
  

Image 2 : Original Difference Projected

Image 3: Original Difference Projected

Image 4: Original Difference Projected

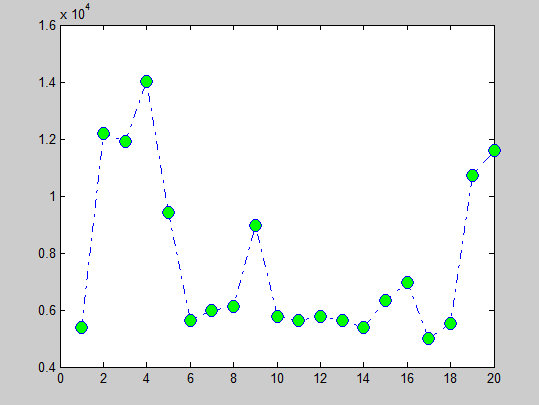
  

Image 5: Original Difference Projected

I have displayed 5 of the non faces images and their result for this answer.

1. Plot for Frobenius Norm Non Face Images



1. Plot for Frobenius Norm for Face Images

