M231 - Pattern Recognition and Machine Learning

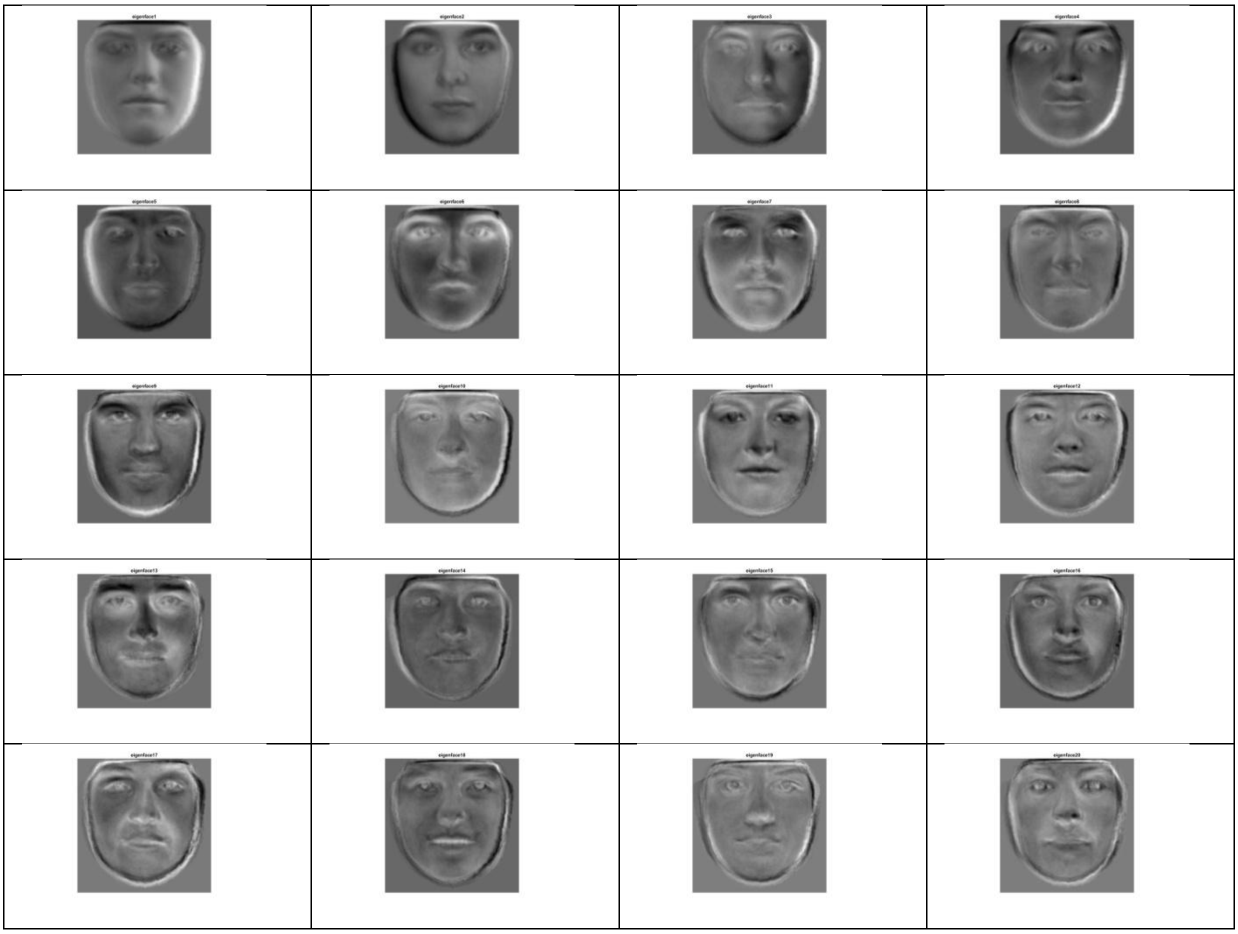
DUCK-HA HWANG
UID#: 404589112

Project I: PCA and FLD for Analyzing Human Faces

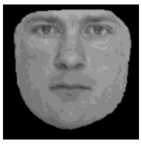
(1). Compute the mean and first k eigenfaces for the training images with no landmark alignment. Display the first K=20 eigenfaces and use them to reconstruct the remaining 27 test faces. Plot the total reconstruction error (squared intensity difference between the reconstructed images and their original ones) per pixel (i.e. normalize the error by the pixel number, and average over the testing images) over the number of eigenfaces k.

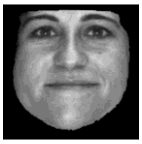
Answer:
MeanFace

Display the first K=20 eigenfaces,
EigenFaces

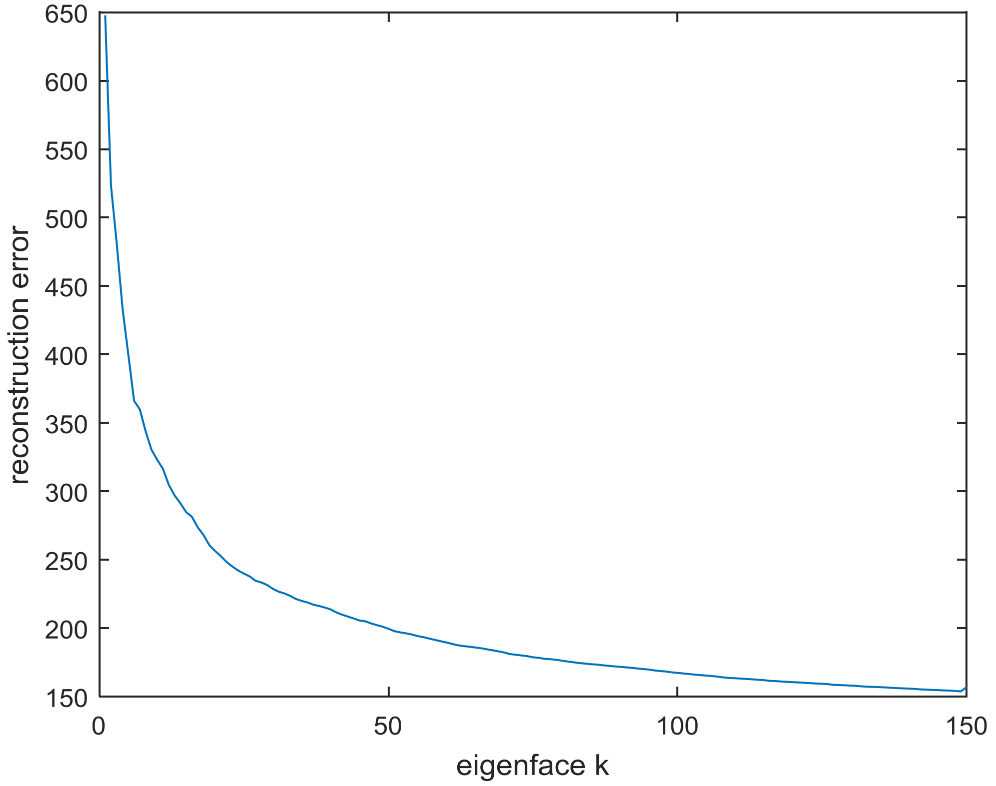


K=20 eigenfaces and use them to reconstruct the remaining 27 test faces

Original test image	Reconstructed test image	Original test image	Reconstructed test image
			
			
			
			
			
			
			
			

Original test image	Reconstructed test image	Original test image	Reconstructed test image
			
			
			
			
			
			

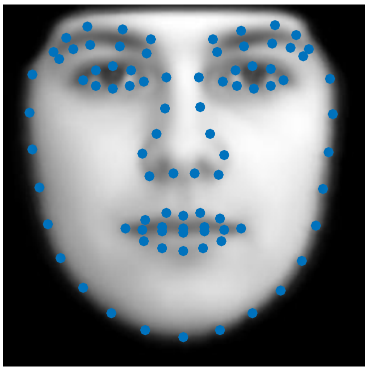
Plot the total reconstruction error per pixel over the number of eigenfaces k.

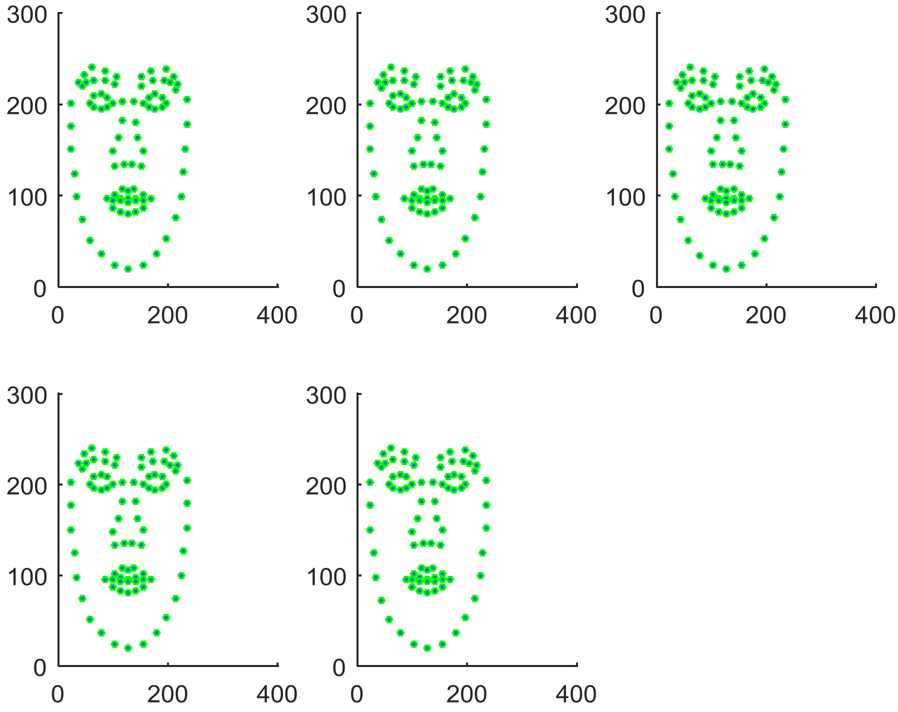


(2). Compute the mean and first k-eigen-warpping of the landmarks for the training faces. Here warping means displacement of points on the face images. Display the first 5 eigen warppings (you need to add the mean to make it meaningful), and use them to reconstruct the landmarks for the test faces. Plot the reconstruction error (in terms of distance) over the number of eigen-warppings k (again, the error is averaged over all the testing images).

Answer:

-Display mean_landmark



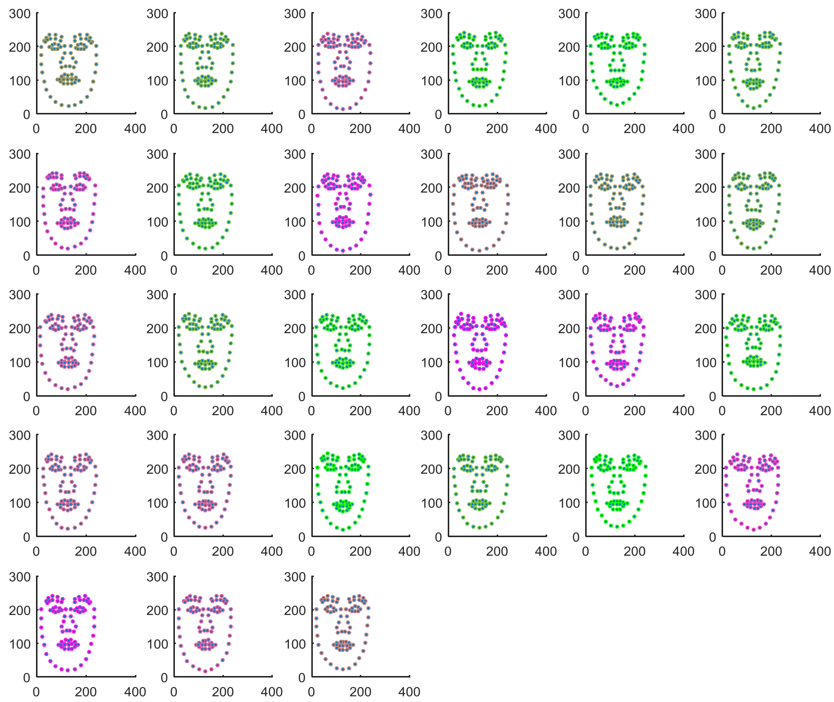


<Explanation>

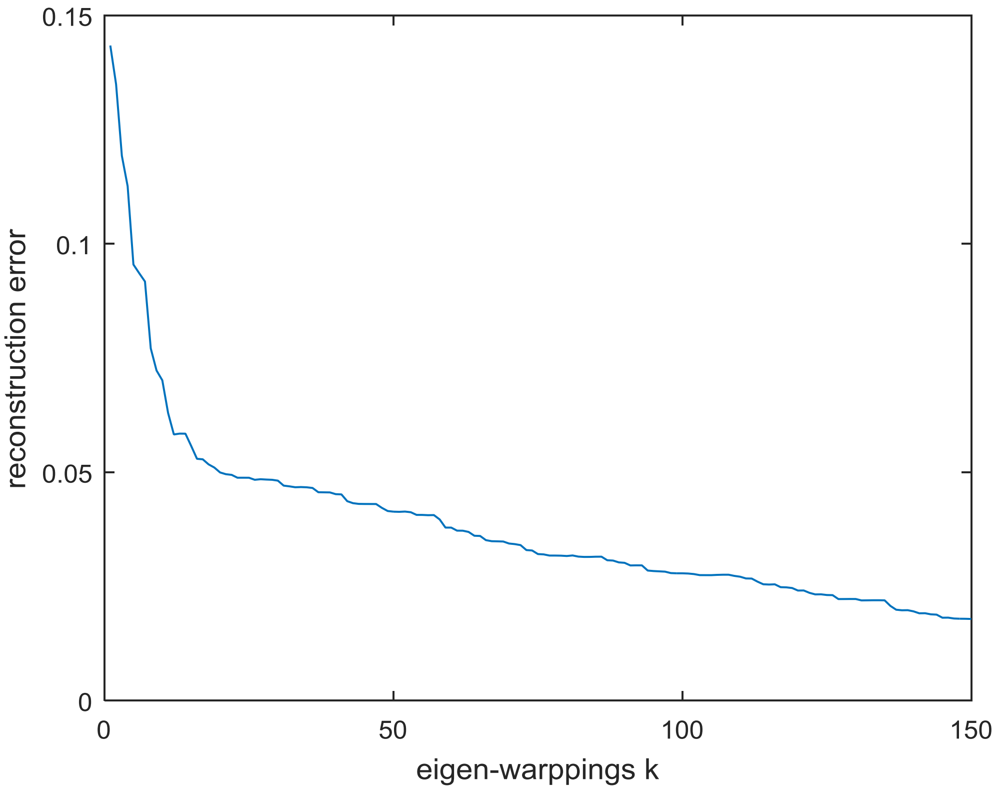
Blue Mark= mean landmarks

Red Mard=first 5-eigen-warpping of the landmarks for the training faces.

-Display the first 5 eigen warppings (you need to add the mean to make it meaningful), and use them to reconstruct the landmarks for the 27 test faces.

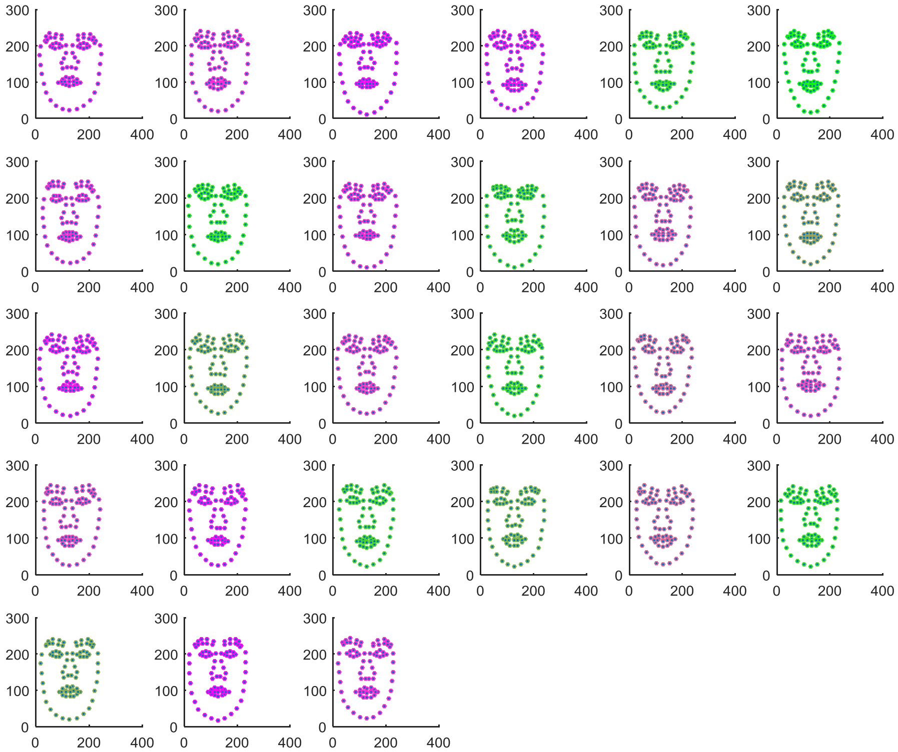


Plot the reconstruction error over the number of eigen-warppings k.



(3). Combine the two steps above. Our objective is to reconstruct images based on top 10 eigenvectors for the warping and then top k (say 10) eigenvectors for the appearance, in the context of compressing the face images and communicate through a network with small number of bits. For the training images, we first align the images by warping their landmarks into the mean position (interpolation between landmarks is needed), and then compute the eigenfaces (appearance) from these aligned images. For each testing face:(i) project its landmarks to the top 10 eigenwarpings, you get the reconstructed landmarks. (Here you lose a bit of geometric precision of reconstruction); (ii) warp the face image to the mean position and then project to the top k (say k=10) eigenfaces, you get the reconstructed image at mean position (here you further lose a bit of appearance accuracy). (iii) Warp the reconstructed faces in step (ii) to the positions reconstructed in step (i). Note that this new image is constructed from 20 numbers. Then compare the reconstructed faces against the original testing images (here you have loss in both geometry and appearance). (iv) Plot the reconstruction errors per pixel against the number of eigenfaces k.

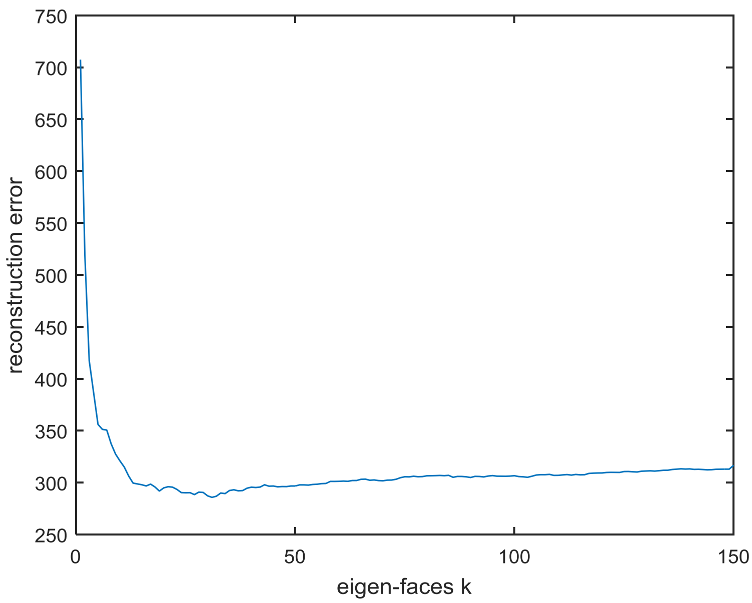
DISPLAY 27 RECONSTRUCTED TEST Warpings(LANDMARKS) FOR K= 10



Display reconstruction of top 27 test faces using Top 10 eigen.
(landmarks Alingned before warping to reconstructed landmarks)

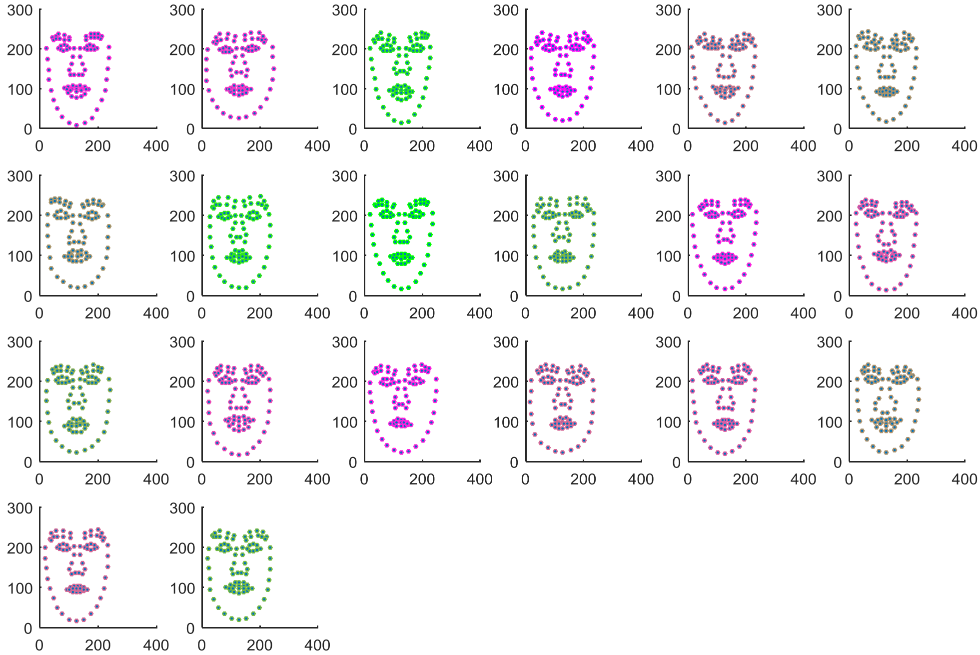
Display reconstruction of top 27 test faces using Top 10 eigen.
(After Warping to reconstructed landmarks)

Plot the reconstruction errors per pixel against the number of eigen-faces k.



(4). Synthesize random faces by a random sampling of the landmarks (based on the top 10 eigen-values and eigen-vectors in the wrapping analysis) and a random sampling of the appearance (based on the top 10 eigen-values and eigenvectors in the intensity analysis). Display 20 synthesized face images. (As we discussed in class, each axis has its own unit, that is, the square root of its eigen-value).

-Synthesize random faces by a random sampling of the landmarks (based on the top 10 eigen-values and eigen-vectors in the wrapping analysis).



Random sampling of the appearance and display 20 synthesized face images.
(Based on the top 10 eigen-values and eigen-vectors in the intensity analysis)

Another set of randomly synthesized 20 face images.

Part 2: Fisher faces for gender discrimination.

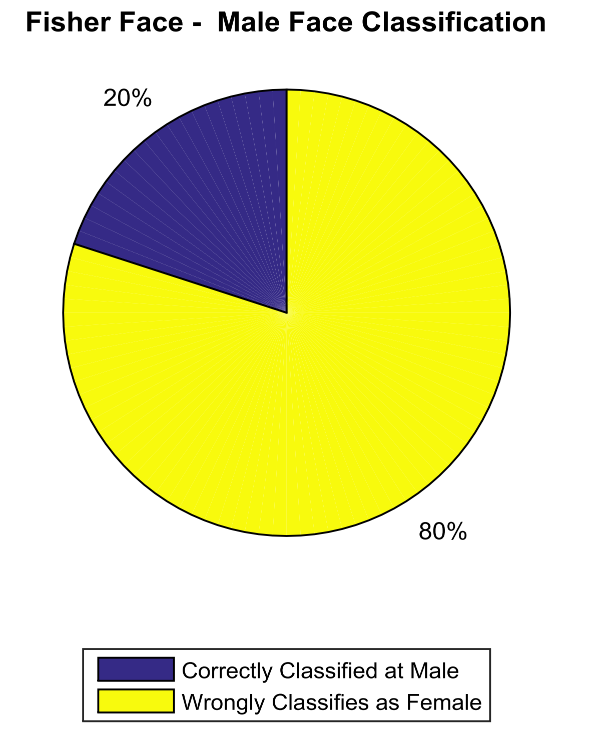
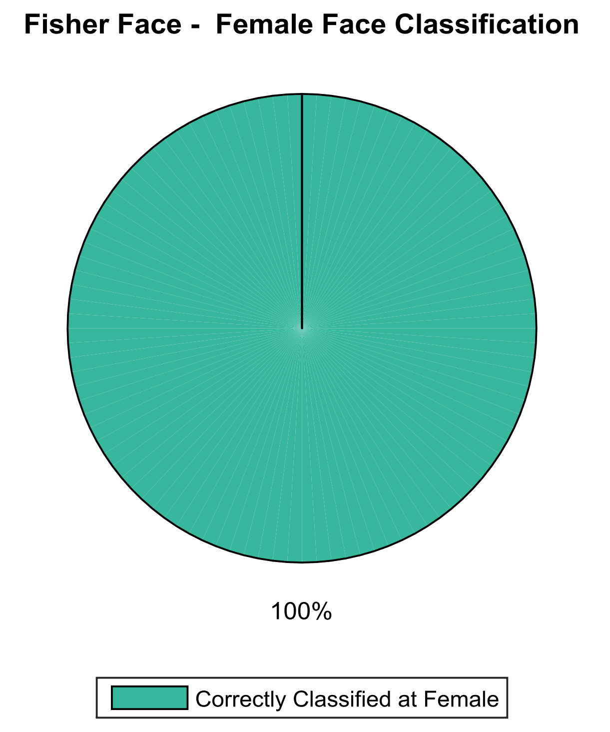
We have divided the 178 faces into male (88) [Face 57 is removed as a duplicate] and female faces (85), plus 4 unknown (which is hard even for human eyes to tell based on the faces alone). Then you choose 10 male and 10 female as the testing set. The remaining faces (except the 4 unknown) will be used as training sets.

(5) Find the Fisher face that distinguishes male from female using the training sets, and then test it on the 20 testing faces and see how much percentage is right. This Fisher face mixes both geometry and appearance difference between male and female.

[The within-class scatter matrix is again very high dimensional, you can either use the trick that we used in (1) for computing its eigen-values and eigenvectors, or you may compute the Fisher faces over the reduced dimensions in steps (2) and (3): i.e. each face is now reduced to 10Dimensional geometric vector + 10 dimensional appearance vector. After the Fisher linear Discriminant analysis, we represent each face by as few as 2 dimensions for discriminative purpose and yet, it can tell apart male from female!

Find the Fisher face that distinguishes male from female using the training sets.
(This Fisher face mixes both geometry and appearance difference between male and female.)

Test it on the 20 testing faces and see how much percentage is right.

	
1) Male's classification success percentage: Answer: 20% x_male: [0.0002 -0.0075 -0.0170 -0.0147 -0.0079 -0.0042 -0.0078 -0.0124 -0.0089 0.0045]	**2) Female's classification success percentage:** Answer: 100% x_female: [0.0157 0.0147 0.0045 0.0078 0.0063 0.0094 0.0021 0.0156 0.0159 0.0223]

(6) Compute the Fisher face for the key point (geometric shape) and Fisher face for the appearance (after aligning them to the mean position) respectively, and thus each face is projected to a 2Dfeature space, and visualizes how separable these points are.

Scatter plot: Appearances and Geometry Test images
Blue: Female, Red: Male