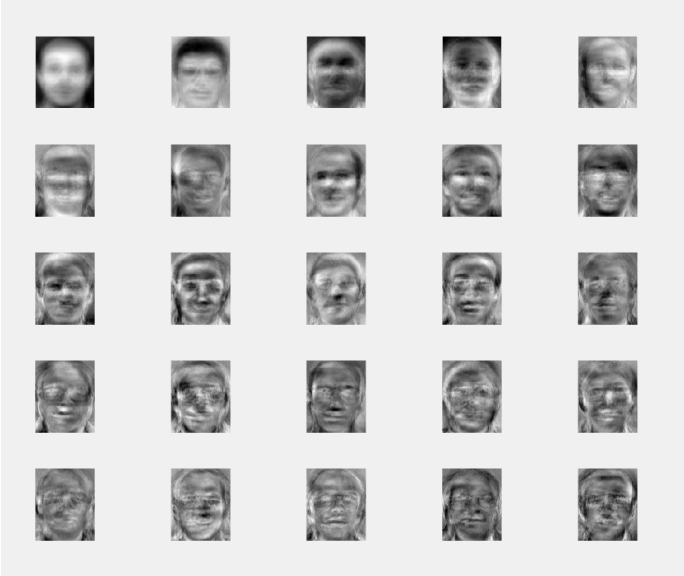
Q5)

Eigenfaces-



These are the eigenfaces that we developed from our training dataset.

As we take the 25 largest eigenvalues, we consider the eigenfaces corresponding to the 25 eigenvectors corresponding to the eigenvalues here.

The eigenface on the bottom right, is the eigenface corresponding to the highest eigenvalue eigenvector.

The eigenfaces along a row correspond to successively increasing eigenvalues.

Similarly eigenvalues increase as we go to the bottom rows.

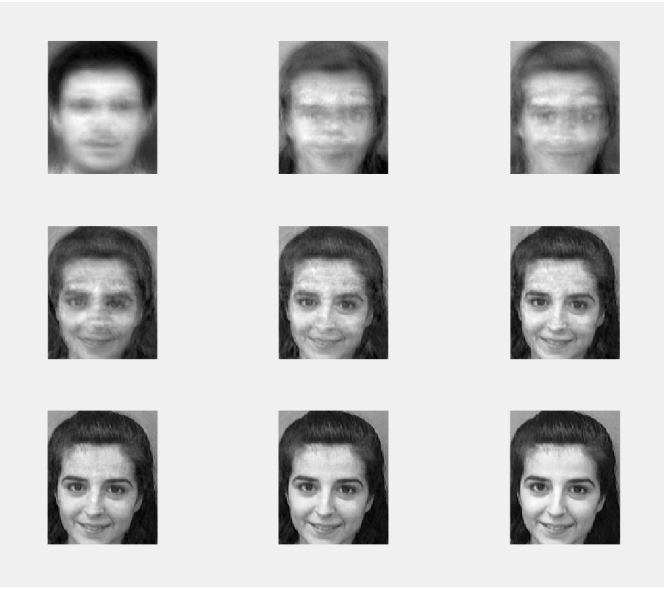
The eigenface corresponding to the 25th largest eigenvalue is the one on the top leftmost.

We see that the lower eigenvalues eigenfaces capture the overall net intensity of vast areas. As our eigenvalues increase, we see that the corresponding eigenfaces become way more detailed and intricate.

Reconstruction-

I have taken person 2, as the person I want to plot the reconstruction for, for all different values of k.

Thus the reconstruction images are as follows-



We start with k=2, and keep increasing it along each row. K also increases as our row number increases, once a single row is over.

We see that for around k=100, we begin to get a very decent reconstruction of the face of the person.

This is as for k's below that, we are not capturing sufficient information by only taking those many largest eigenvectors, hence we are not able to reconstruct the face very accurately.

We also see that as our k increases beyond a certain point, there is not too much of a difference in the reconstructed faces. This is as, almost all of the information has already been retained by the larger eigenvalues eigenvectors, before that particular eigenvalue, hence the reconstructed faces look fairly similar.

Code-

```
% INITIALIZATION

image_size = 112*92;
train_image_array = zeros(image_size,32,6);
test_image_array = zeros(image_size,32,4);
not_in_gallery_image_array = zeros(image_size,8,4);
K = [1 2 3 5 10 15 20 30 50 75 100 150 170];
final_rr = zeros([length(K),1]);

%FILE READING
directory_name = dir('ORL/');
is_directory = [directory_name.isdir]  & ~strcmp({directory_name.name},'.') &
~strcmp({directory_name.name},'..');
person_directory = directory_name(is_directory);

for i= 1:length(person_directory)
    image_files = dir(['ORL/' person_directory(i).name '/*.pgm']);
    for j=1:length(image_files)
        image = double(imread(['ORL/' person_directory(i).name '/' image_files(j).name]));
        if (i > 32) && (j<=4)
            not_in_gallery_image_array(:,i-32,j) = image(:);
        elseif (i <= 32) && (j>6)
            test_image_array(:,i,j-6) = image(:);
        elseif (i <= 32) && (j<=6)
            train_image_array(:,i,j) = image(:);
        end
    end
end

train_image_array = reshape(train_image_array,image_size,[]);
test_image_array = reshape(test_image_array,image_size,[]);
not_in_gallery_image_array = reshape(not_in_gallery_image_array,image_size,[]);
%imshow(reshape(train_image_array(:,1),112,92),[])

%MEAN
mean_images = mean(train_image_array(:,:),'all');
train_image_array= train_image_array-mean_images;

%SVD
[U,S,V] = svd(train_image_array);



%ORL FACE Reconstruction
K_r = [2 10 20 50 75 100 125 150 175];
h =[];
for i=1:length(K_r)
  U_k = U(:,1:K_r(i));
  projected_image = transpose(U_k)*train_image_array(:,2);
  recovered_image = U_k*projected_image;
  subplot(3,3,i),imshow(reshape(recovered_image,112,[]),[])
```

```
end

%25 Eigenfaces
%Eigenfaces = U(:,1:25);
%for i = 1:25
%    subplot(5,5,i),imshow(reshape(Eigenfaces(:,i),112,[]),[])
%end
```