PCA : Face Recognition

Rahul Garg 2003CS10183 Varun Gulshan 2003CS10191

October 29, 2005

Overview

PCA tries to identify the directions of maximum variation in the training data. Using the SVD decomposition,we can find the first k-vectors whose singular values are the largest. We basically take the best projection of the training images onto a subspace. The images when represented in this subspace have maximum variance(in the coordinates).

Advantages of PCA

1. Smaller representation of database because we only store the training images in the form of their projections on the reduced basis.

2. Noise is reduced beacuse we choose the maximum variation basis and hence features like background with small variation are automatically ignored.

Procedure

To identify an input image we proceed as follows

1. Evaluate the components of input image along the selected k eigen vectors.

2. Reconstruct the image from the components.

3. If the distance between the reconstructed image and the original image is above a threshold ϵ, the the input image is not a face image. We tried this threshold but it did not turn out to be a reliable estimate.

4. Now compute the distance of the input image from the training images in the space spanned by the k eigen vectors.If the minimum distance is above threshold ϵ then the input image is not a face from the training database else report the training image with the minimum distance as the recognized image.

The threshold ϵ is defined as

$$\epsilon = \frac{1}{2} max\{\|d_{ij}\|\}$$

where d_{ij} denotes the distance between training images i and j.

Instead of working with individual images, we can cluster the images of an individual in a single class. Distance to a class may be defined as average of the distances to each of the individual images within the class.

Observations

Broadly we can divide the input into three types

1. Face of a person whose training image has been provided in the training database
2. Face of new person
3. An arbitrary image(not a face)

For each of the three classes for inputs we can monitor two types of distances

1. Distance between the reconstructed image and the original image.
2. Distance to the nearest image in the training database.

The data is presented in the following table

No of Evecs	Dist to training image	Dist to reconstructed image
Face belonging to training set		
5	602	3241
10	1089	3031
20	1263	2768
30	1547	2532
40	1609	2396
New face		
5	1255	3147
10	2308	2925
20	2951	2783
30	3107	2664
40	3178	2614
Arbitrary Image(Non Face)		
5	1062	10515
10	3039	10046
20	4652	9574
30	6014	8759
40	6197	8673

Some observations may be made

1. Distance to the reconstructed image decreases with no of eigenvectors considered. This is expected because we project onto a higher dimensional space and hence the residual error is reduced. Similar arguement also explains the increase in distance to the nearest training image.

2. New Face vs Trained Face
   If we consider the distances to the reconstructed image , there is not much distinction to be made. But these two types of images may be distinguished by comparing the distances to the nearest image in the training image database. Hence we can use a suitable threshold to distinguish between the two.

3. Face image vs Arbitrary Image
   These two types of images may be distinguished by comparing the distances to the reconstructed image. The eigen space for face image will not be a proper representation for an arbitrary image and hence distance will be larger in this case(of the order of 10^4).

To investigate the proper choice of number of eigenvectors we can also analyse the following data.The following table displays the 5 nearest matches to an input face image(trained) with a $\surd$ denoting a correct match and $\emptyset$ denoting a false match.

No of Evecs	Match 1	Match 2	Match 3	Match 4	Match 5
5	$\surd$	$\surd$	$\emptyset$	$\emptyset$	$\surd$
10	$\surd$	$\surd$	$\emptyset$	$\surd$	$\surd$
20	$\surd$	$\surd$	$\emptyset$	$\surd$	$\surd$
30	$\surd$	$\surd$	$\surd$	$\surd$	$\emptyset$
40	$\surd$	$\surd$	$\surd$	$\surd$	$\emptyset$

From the above table we see that the wrong match has been pushed to the end by the time we reach 30th eigen vector. Note that two wrong matches are returned if we consider only 5 eigenvectors. From the previous table also, if we carefully observe the distances to the nearest training image for a new face and an arbitrary image, we observe that we can clearly distinguish between the two types of images if we use 20 or 30 eigen vectors. Hence a choice of 30 eigen vectors seems optimal. We can also see it from the plot of eigenvalue magnitudes.

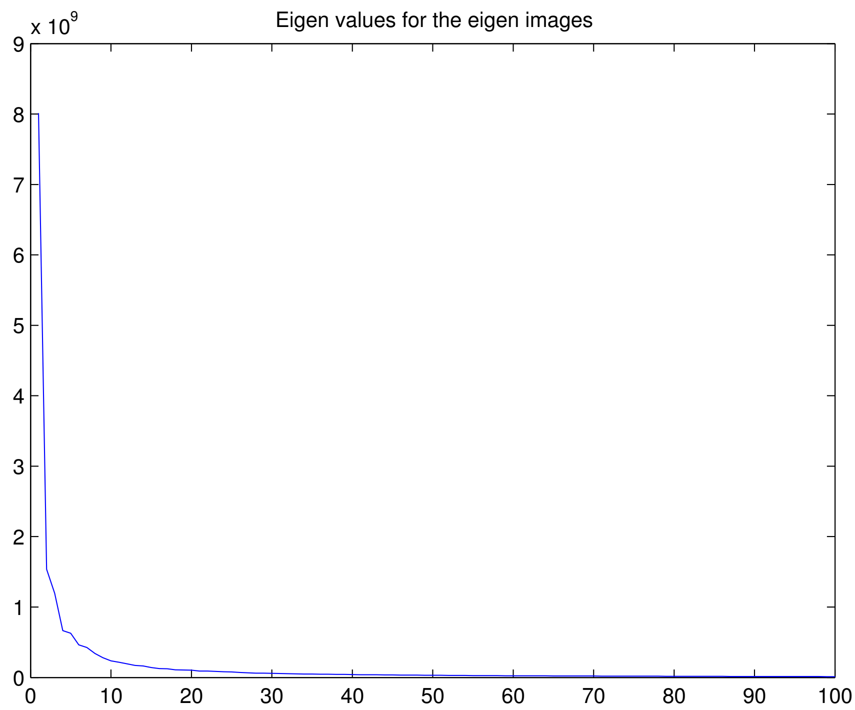

Figure 1: Eigen value magnitudes, After 30 the magnitude falls off rapidly

Results

For our database of size 200, the mean image is shown below

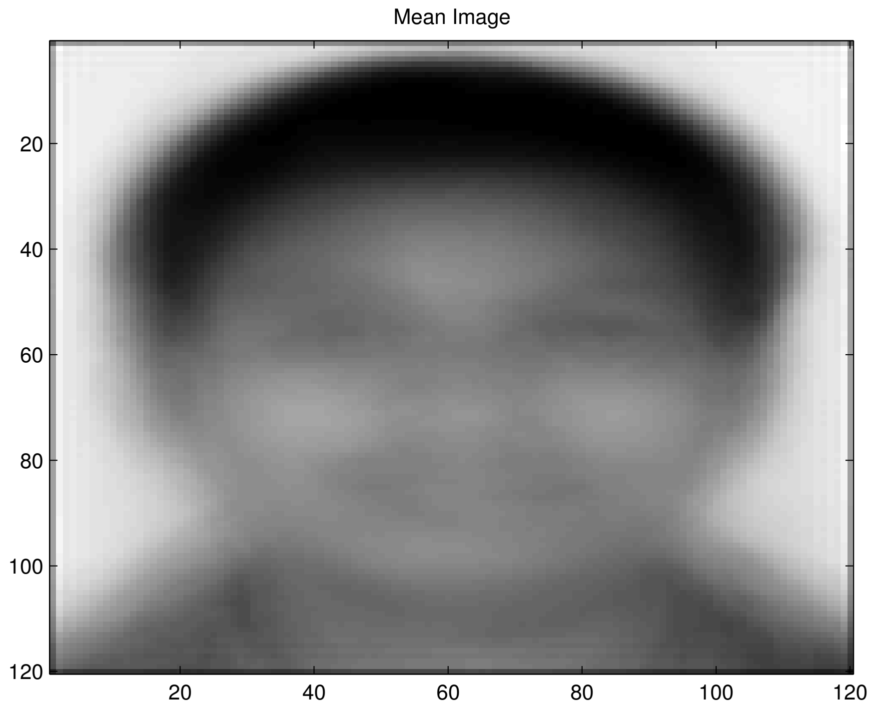

Figure 2: Mean image

The first 5 eigenfaces are also displayed

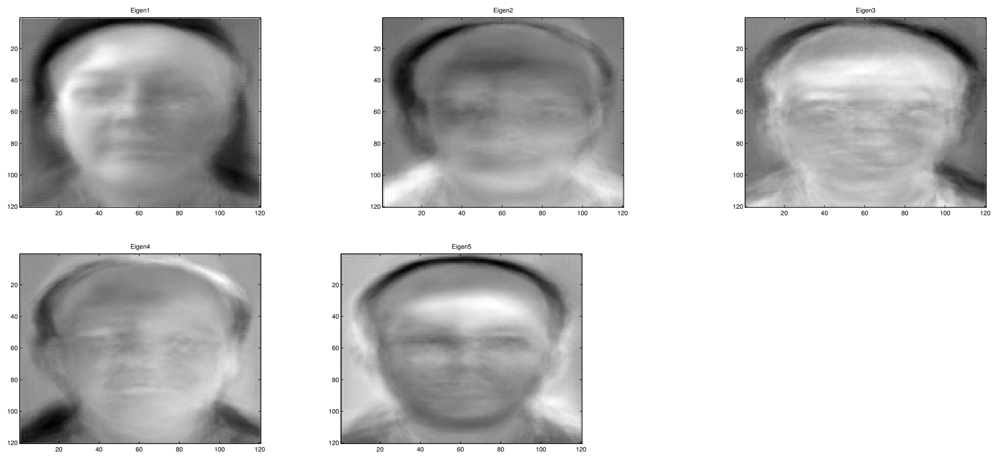

Figure 3: First five eigen faces

The following figure displays the result of an example query

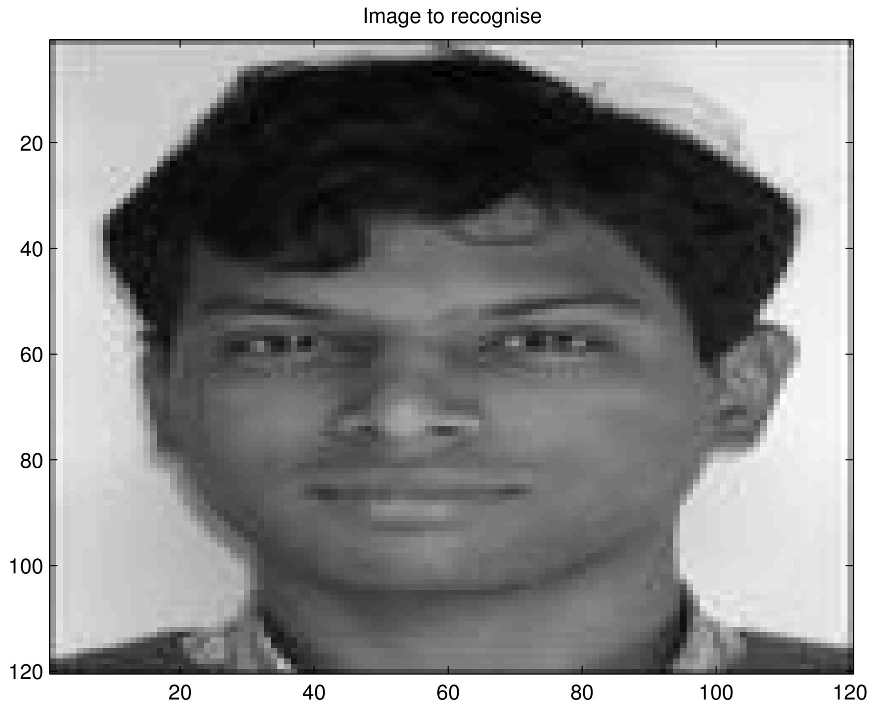

Figure 4: Face to recognise

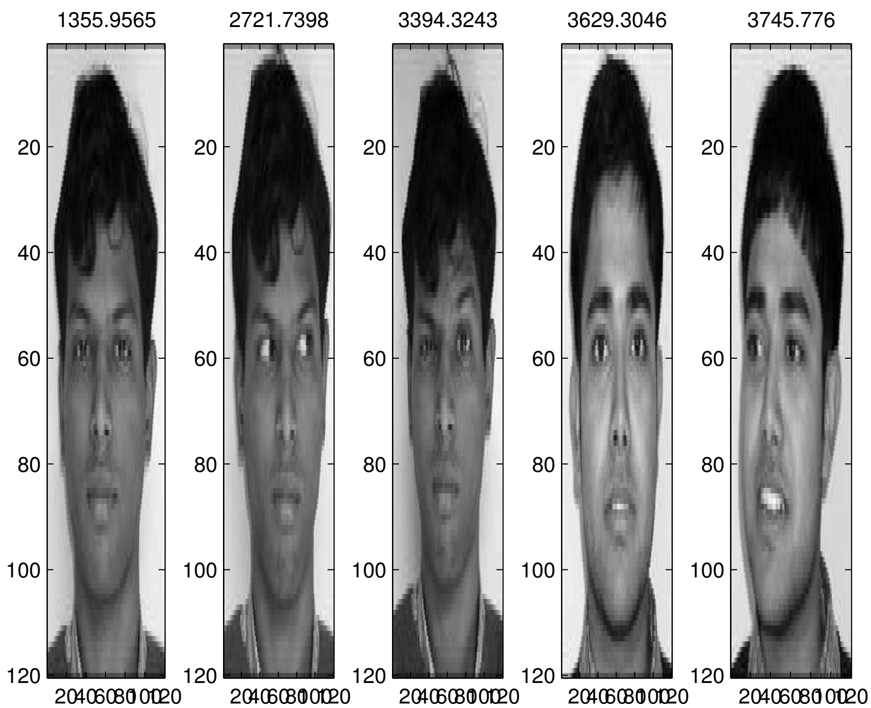

Figure 5: Matches found