ES-331 Probability and Random Process

Assignment - 2

Image Recognition using Eigenfaces

Brief Report

Name: Karan Bhardwaj
Roll No.:20110093

Introduction:

Face Recognition using Eigenfaces is a face recognition technique that uses the concept of linear algebra and dimension reduction with the help of computer vision. Eigenfaces are a nonlinear face detection method that uses singular value decomposition to perform the calculation. The idea behind eigenfaces is to extract characteristics of a face, like its eyes and smile, by looking at the individual components that make up a face in terms of its shape.

The algorithm that involves the use of Eigenfaces and involves dimension reduction of the eigenspace is the PCA (Principle Component Analysis). The model I will create for face recognition will involve using PCA.

Task:

To design a face-recognition model using the eigenfaces method. The model should detect whether a given person's image is present in the dataset or not and, if present, determine the number of images matched.

Training Algorithm:

(PCA - Principle component Analysis)

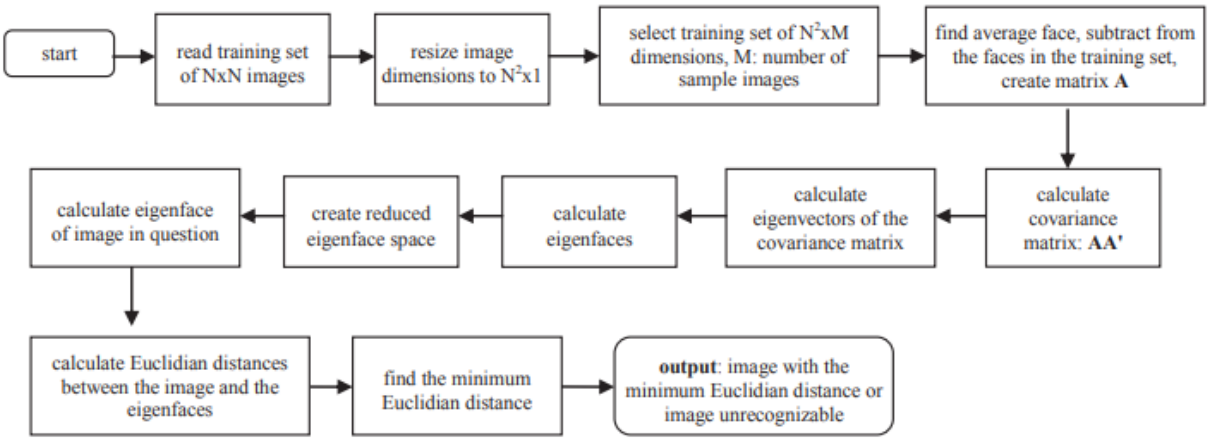


Fig. showing the complete methodology of Image Recognition using the eigenfaces method

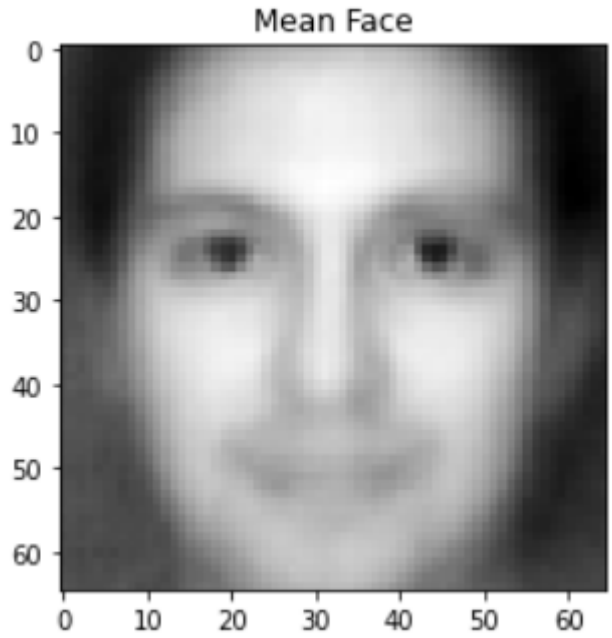
The above flowchart shows the steps involved in the Principle component analysis algorithm.

Procedure:

- The first step is to take the image dataset.
- Extract the faces (used haar_face to extract) from the images and resize all the faces to size (N, N).
- Put all the images in a NumPy array and apply the PCA algorithm as described above.
- Obtain the mean face using **np.mean(),** covariance matrix using **np.cov()**, Eigenvectors and Eigenvalues using SVD(singular value decomposition).
- Further, obtain the eigenfaces and perform image reconstruction.
- Finally, take the test image as input and run the **findingmatch()** function to find out whether the test face matches with dataset faces.
- If there is no matching, the output will be as "No Match Found", else "No. of Match found: _" will be displayed along with the images found.

Results:

Mean Face:



Eigenfaces:

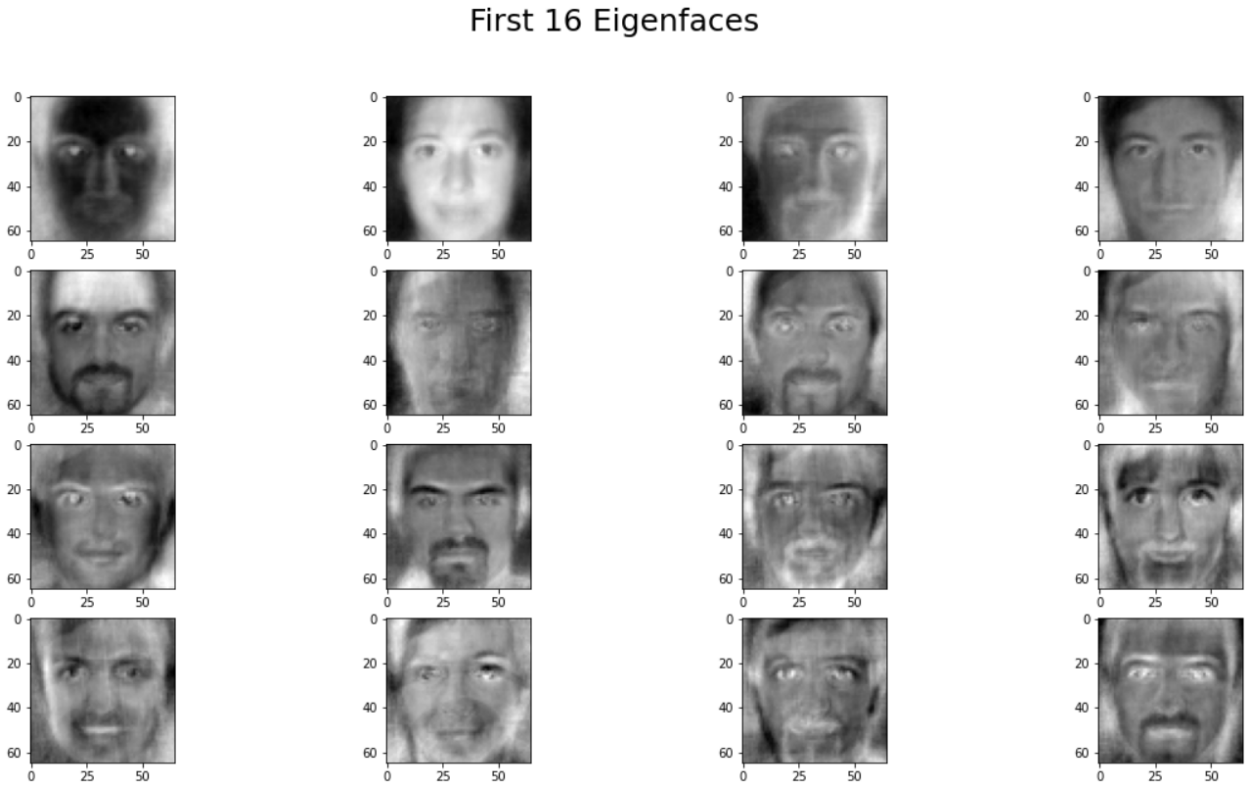
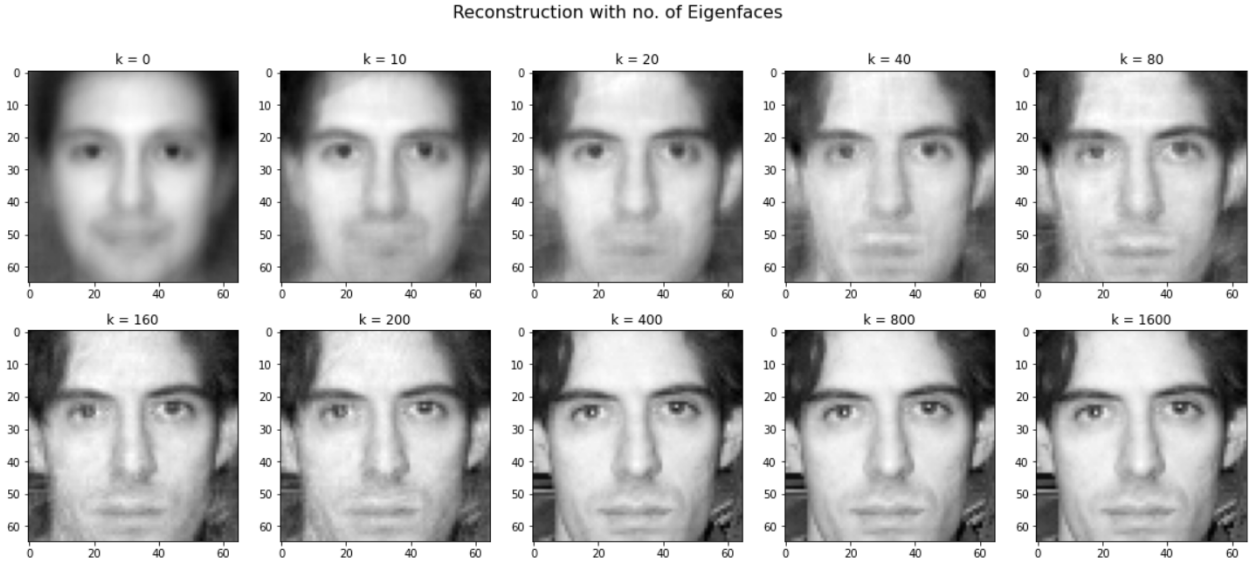
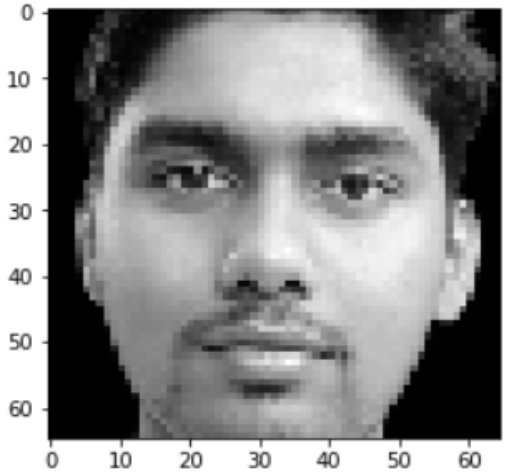


Image Reconstruction:



Test Images Output:

Using my own image as a test face:



Output:

```
def findingmatch(test_img,dataset):
  arr = []
  for i in range(450):
    weight = faces_norm[i,:].dot(eigen_vecs[:,:800])
    projected_face = weight.dot(eigen_vecs[:,:800].T)
    data_img = projected_face.reshape(65,65)+mean_face_mat.reshape(65,65)

    ed = euc_dist(test,data_img) # Measuring the eucleadian distance
    # Applying threshold
    if(ed < 1500):
      # print(mse)
      arr.append(data_img)

  if(len(arr)==0):
    print("No Match Found")
  else:
    print("No. of faces matched: ",len(arr))
  return arr
```

```
[80]
matched_img = findingmatch(test,img_array)
n = len(matched_img)
for i in np.arange(n):
  axis  = plt.subplot(1,n,i+1)
  img = matched_img[i]
  plt.imshow(img, cmap = 'gray')
```

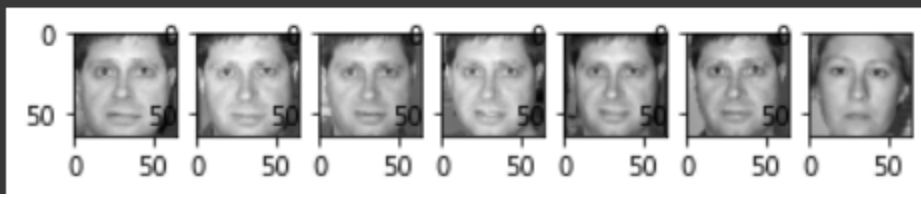
```
No Match Found
```

Using One of the dataset faces as a test face:

Output:

```
matched_img = findingmatch(test,img_array)
n = len(matched_img)
for i in np.arange(n):
  axis  = plt.subplot(1,n,i+1)
  img = matched_img[i]
  plt.imshow(img, cmap = 'gray')
```

```
No. of faces matched:  7
```



Conclusion:

From the above result, I find found that when I used my image as the test face, no faces matched, which gives a good indication of the model working effectively.

Also, when the above dataset image is taken as a test face, it can be seen that 7 images of the dataset matched, out of which 6 can be seen as the same as the input image face.
Similar observations were observed when other dataset images were used as test faces, meaning out of all the faces matched, not all were the same as the test face.
The accuracy of the model came out to be around 55 to 60%. Further accuracy can be improved by using better face extraction techniques from the images.

It can be concluded that the face recognition model using eigenfaces recognizes faces that are present in the dataset but not outside the dataset.

Advantages and Disadvantages:

The disadvantages and advantages of the above-performed Image recognition technique are as follows:

Advantages:

1. Easy to implement and computationally requires less time.
2. No knowledge of machine learning is required.

Disadvantages:

1. Faces need to be strictly aligned.
2. The algorithm is sensitive to lighting, shadows and the face's scale.
3. Faces are needed to be extracted from the image properly.

Reference:

The following resources have been considered while applying the PCA algorithm for finding the eigenfaces:

1. https://www.geeksforgeeks.org/ml-face-recognition-using-eigenfaces-pca-algorithm/
2. https://pyimagesearch.com/2021/05/10/opencv-eigenfaces-for-face-recognition/
3. https://www.sciencedirect.com/science/article/pii/S2212017312000242?ref=pdf_download&fr=RR-2&rr=75b2619a8f268a20