USING PCA FOR DIMENSIONALITY REDUCTION OF FACIAL IMAGES

Susmitha Neerubai (C534C335)
Department of Computer Science and Networking
Wichita state university, Wichita, Kansas – 67260, U.S.A.

***Abstract* - Face Recognition is a typical issue in Machine Learning. This innovation has just been generally utilized in our lives to ensure private security. Face pictures accompanies diverse foundation, variation light, unique outward appearance and impediment. One of the fundamental face acknowledgment systems is eigenface which is very basic, efficient, and yields commonly great outcomes in controlled conditions. Along these lines, this paper exhibits an exploratory presentation examination of face acknowledgment utilizing Principal Component Analysis (PCA) and Normalized Principal Component Analysis (NPCA).**

In this paper, we discuss how Principal Component Analysis (PCA) works, and how it can be used as a dimensionality reduction technique for classification problems. At the end of this article, MATLAB source code is provided for demonstration purposes. The least discriminative features can be found by a certain various and greedy feature selection approaches. However, in practice, many features depend on each other or on an underlying unknown variable. A single feature could therefore represent a combination of multiple types of information by a single value. Removing such a feature would remove more information than needed. In the next paragraphs, we introduce PCA as a feature extraction solution to this problem. We use PCA on a simulated dataset. The number of principal components *k* should be determined based on proportion of variance. The eigen subspace consist of chosen eigenvectors.

Keywords: PCA – Principal component analysis, Dimensionality reduction, face recognition.

1. INTRODUCTION:

Dimensionality Reduction:

Dimensionality Reduction is the process of reducing the number of random variables under consideration by obtaining a set of principal variables. Approaches can be divided into feature selection and feature extraction.

Face recognition has a challenge to perform in real time. Raw face image may consume a long time to recognize since it suffers from a huge number of pixels. One needs to reduce the amounts of pixels. This is called dimensionality reduction or feature extraction, to save time for the decision step. Feature extraction refers to transforming face space into a feature space. In the feature space, the face database is represented by a reduced number of features that retain most of the important information of the original faces and becomes easier to visualize the data when reduced to very low dimensions such as 2D or 3D.

The advantage of this reduction in dimensions is that it removes information that is not useful and specifically decomposes the structure of face into components which are uncorrelated and are known as Eigen faces. The major benefit of this method is that it can trim down the data required to recognize the entity to 1/1000th of the data existing. Suppose, we would like to perform face recognition, i.e. determine the identity of the person depicted in an image, based on a training dataset of labelled face images. One approach might be to treat the brightness of each pixel of the image as a feature. If the input images are of size 32×32 pixels, this means that the feature vector contains 1024 feature values. Classifying a new face image can then be done by calculating the Euclidean distance between this 1024-dimensional vector, and the feature vectors of the people in our training dataset. The smallest distance then tells us which person we are looking at.

However, operating in a 1024-dimensional space becomes problematic if we only have a few hundred training samples. Furthermore, Euclidean distances behave strangely in high dimensional spaces. Therefore, we could use PCA to reduce the dimensionality of the feature space by calculating the eigenvectors of the covariance matrix of the set of 1024-dimensional feature vectors, and then projecting each feature vector onto the largest eigenvectors.

2. METHODS:

PCA ALGORITHM:

- Select an image on which the analysis needs to be performed. Read the image in the MATLAB and get the data in form of matrix.
- Give that data as the input to the MATLAB and perform the code on it.
- Compute the mean of the data matrix "The mean of each row"
- Subtract the mean from each image [Centring the data]
- Compute the covariance matrix (co)
- Compute the eigen values and eigen vectors of the covariance matrix
- Sort the eigen vectors according to the eigen values
- Compute the number of eigen values that greater than zero (you can select any threshold)

Then we print the eigen vectors and then find the trends by selecting different value for principal components.

Mathematics for PCA:

The images are mean centred by subtracting the mean image from each image vector. Let 'm' represent the mean image.

$$m = \frac{1}{M}\sum_{i=1}^{M} x_i$$

Let, W_i be defined as mean centered image $W_i = X_i - m$

A covariance matrix is constructed as $C = AA^T$

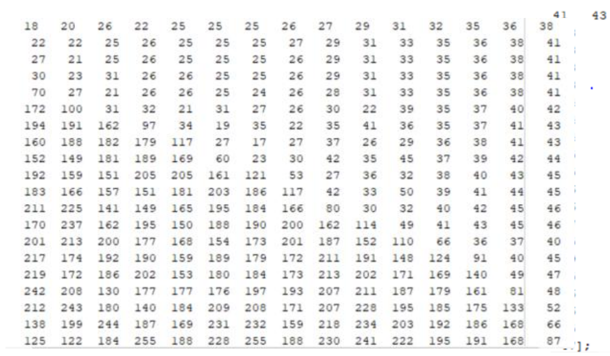
Where, $A = [\Phi_1,\Phi_2,\Phi_3,..,\Phi_M]$ of size $N^2 \times N^2$

3. MATLAB CODE:

```
data=imread('e:\Users\sush\Downloads\sush.jpg');
[r,c]=size(data);
m=mean(data')';
d=data-repmat(m,1,c);
co=d*d';
[eigvector,eigvl]=eig(co);
eigvalue = diag(eigvl);
[junk, index] =
sort(eigvalue,'descend');
eigvalue = eigvalue(index);
eigvector = eigvector(:,index);
count1=0;
for i=1:size(eigvalue,1)
    if(eigvalue(i)>0)
    count1=count1+1;
    end
end
vec=eigvector(:,1:count1);
x=vec'*d;
eigvalue
total=sum(eigvalue);
total
```

4. RESULTS AND DISCUSSION:

Data:



18	20	26	22	25	25	25	26	27	29	31	32	35	36	38
22	22	25	26	25	25	25	27	29	31	33	35	36	38	41
27	21	25	26	25	25	25	26	29	31	33	35	36	38	41
30	23	31	26	26	25	25	26	29	31	33	35	36	38	41
70	27	21	26	26	25	24	26	28	31	33	35	36	38	41
172	100	31	32	21	31	27	26	30	22	39	35	37	40	42
194	191	162	97	34	19	35	22	35	41	36	35	37	41	43
160	188	182	179	117	27	17	27	37	26	29	36	38	41	43
152	149	181	189	169	60	23	30	42	35	45	37	39	42	44
192	159	151	205	205	161	121	53	27	36	32	38	40	43	45
183	166	157	151	181	203	186	117	42	33	50	39	41	44	45
211	225	141	149	165	195	184	166	80	30	32	40	42	45	46
170	237	162	195	150	188	190	200	162	114	49	41	43	45	46
201	213	200	177	168	154	173	201	187	152	110	66	36	37	40
217	174	192	190	159	189	179	172	211	191	148	124	91	40	45
219	172	186	202	153	180	184	173	213	202	171	169	140	49	47
242	208	130	177	177	176	197	193	207	211	187	179	161	81	48
212	243	180	140	184	209	208	171	207	228	195	185	175	133	52
138	199	244	187	169	231	232	159	218	234	203	192	186	168	66
125	122	184	255	188	228	255	188	230	241	222	195	191	168	87

Figure 3data obtained from the image

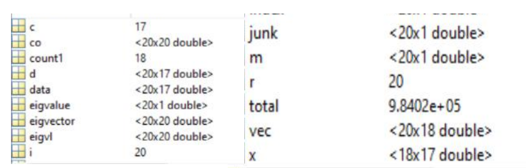


Figure 1result of the matlab code

Figure 2 the eigen values

Compute the Propotion of variance for different k values:

O/P = sum (K number of eigen values)/ Total sum of the eigen values obtained

If K=2 , O/P= (EV1+EV2)/Total sum of the eigen values where EV1- eigen value 1 & EV2- eigen value2

K value	Propotion of variance
2	0.791
5	0.953
8	0.99
11	0.99

Discussions:

WE do not see much variation but the trend is that as the K value keeps on increasing the propotion of variance also increases and then becomes constant at a point and stabilizes.

The dimentionality reduces , the output of the pca analysis is the eigen values that we have obtained after the execution.

CONCLUSION:

In this paper we implemented the face recognition system using Principal Component Analysis and Eigenface approach. The system successfully reduced the dimensions for the analysis of the human faces and worked better in different conditions of face orientation.

In this research, Principal component analysis approach to the face recognition problem was studied and a face recognition system based on the eigenfaces approach was proposed. The algorithm has been tested for the image database and implemented using MATLAB. The algorithm developed in a generalized one which works well with any type of images. The tests conducted on PNG images and JPEG images of various subjects in different poses showed that this method gave very good classification of faces though it has limitations over the variations in size of image. The eigenface approach thus provides a practical solution that is well fitted to the problem of face recognition. It is fast, relatively simple and has been shown to work well in constrained environment.

REFERENCES:

[1] Face Recognition Machine Vision System Using Eigenfaces Fares Jalled, Moscow Institute of Physics & Technology, Department of Radio Engineering & Cybernetics, May 2017.

[2]Jure Leskovec, Mining of Massive datasets,section 11.1,page405

[3] [1] Hiremath, Vinay, and Ashwini Mayakar. ”Face recognition using Eigenface approach.” IDT workshop on interesting results in computer science and engineering, Sweden. 2009.

[4] Face Recognition using Principle Component Analysis Kyungnam Kim Department of Computer Science University of Maryland, College Park MD 20742, USA.

[5]online available : https://blog.cordiner.net/2010/12/02/eigenfaces-face-recognition-matlab/