Computer and Communication Engineering Program
Faculty of Engineering
Alexandria University

CSE-357 Pattern Recognition

Face Recognition with Principal Component Analysis (PCA)

Name	ID
Omar Ibrahim Elsayed	7442
Mazen Nagy Mansour	7475

Why is PCA used in Face Recognition?

In face recognition, images of faces are typically high-dimensional data, with each pixel representing a feature. These high-dimensional data can be computationally expensive to work with and may lead to overfitting. PCA helps reduce the dimensionality of the data by transforming it into a lower-dimensional space while retaining as much variance as possible. This reduction in dimensionality simplifies subsequent processing steps and improves computational efficiency.

Data Set

The dataset contains images from 40 individuals, each of them providing 10 images. The pixel intensities of the 400 face images will be used training and testing our face recognition model.

Procedures

1. Read images from files and reshape its vector representation from 112 x 92 to 1 x 10304.

```
Reading Started
Reading Finished
Number of Images loaded:  400
Size of Vector per Image:  10304
```

Fig. 01: Reading Data Set

2. Split the produced data matrix into two submatrices:
   A. Two equally sized matrices each of size 200 x 10304 which corresponds to taking 5 images for each person as training samples and 5 as testing samples.

```
Train Data
      0      1      2      3      4      5      6      7      8      9    ...  \
1    48     49     45     47     49     57     39     42     53     49   ...
1    39     44     53     37     61     48     61     45     35     40   ...
1    64     76     80     53     34     72     60     66     66     50   ...
1    41     47     47     46     44     49     48     58     61     49   ...
1    42     41     44     46     48     39     37     37     33     37   ...
..   ...    ...    ...    ...    ...    ...    ...    ...    ...    ...  ...
40   119    121    123    121    120    123    121    120    123    124  ...
40   131    125    126    131    125    129    125    127    127    127  ...
40   128    125    125    129    128    132    125    133    125    131  ...
40   129    127    133    124    131    129    130    129    127    132  ...
40   119    120    120    118    120    121    121    116    120    121  ...

    10294  10295  10296  10297  10298  10299  10300  10301  10302  10303
1      39     44     40     41     49     42     44     47     46     46
1      23     30     36     32     28     32     31     29     26     29
1      31     28     34     32     35     34     35     35     37     39
1      27     34     35     34     36     34     39     35     37     38
1      42     33     39     39     40     41     43     42     43     41
..    ...    ...    ...    ...    ...    ...    ...    ...    ...    ...
40     34     34     34     38     39     35     36     36     40     36
40     91     89     90     90     93     88     93     89     93     91
40     87     86     87     92     88     85     91     85     90     84
40     91     92     93     90     90     92     89     93     93     93
40     63     88     87     97    101     91     93     89     94     85

[200 rows x 10304 columns]
```

Fig. 02: Train Data 5-5 Data Split

```
Test Data
     0     1     2     3     4     5     6     7     8     9     ...  \
1     60    60    62    53    48    51    61    60    71    68  ...
1     63    53    35    36    33    34    31    35    39    43  ...
1     43    50    41    58    78    83    67    48    44    46  ...
1     44    43    32    32    30    30    38    40    48    66  ...
1     34    34    33    32    38    40    39    49    54    57  ...
..   ...   ...   ...   ...   ...   ...   ...   ...   ...   ...  ...
40   127   131   128   129   127   128   127   128   128   129  ...
40   130   123   127   125   126   126   127   125   125   127  ...
40   123   121   126   122   127   127   123   124   123   127  ...
40   125   119   124   125   124   121   123   125   123   123  ...
40   125   124   124   126   123   125   127   123   124   124  ...

    10294  10295  10296  10297  10298  10299  10300  10301  10302  10303
1      27     35     28     33     31     31     37     32     34     34
1     173    169    166    161    158    169    137     41     10     24
1     167    164    164    162    159    156    155    158    153    169
1      37     42     42     40     33     29     37     43     43     37
1      42     44     38     30     37     30     36     37     40     33
..    ...    ...    ...    ...    ...    ...    ...    ...    ...    ...
40     91     90     87     91     90     90     89     94     84     89
40     39     34     33     37     34     38     41     33     37     40
40     29     47     34     36     42     34     39     40     35     42
40     40     34     38     37     32     41     40     36     39     40
40     39     34     32     30     38     27     36     36     35     34

[200 rows x 10304 columns]
```

Fig. 03: Test Data 5-5 Data Split

B. Two matrices one of size 280 x 10304 for training samples and another of size 120 x 10304 for testing samples which corresponds to taking 7 images for each person for training and 3 for testing.

```
Train Data
     0     1     2     3     4     5     6     7     8     9     ...  \
1     48    49    45    47    49    57    39    42    53    49  ...
1     60    60    62    53    48    51    61    60    71    68  ...
1     63    53    35    36    33    34    31    35    39    43  ...
1     64    76    80    53    34    72    60    66    66    50  ...
1     41    47    47    46    44    49    48    58    61    49  ...
..   ...   ...   ...   ...   ...   ...   ...   ...   ...   ...  ...
40   130   123   127   125   126   126   127   125   125   127  ...
40   128   125   125   129   128   132   125   133   125   131  ...
40   129   127   133   124   131   129   130   129   127   132  ...
40   125   119   124   125   124   121   123   125   123   123  ...
40   125   124   124   126   123   125   127   123   124   124  ...

    10294  10295  10296  10297  10298  10299  10300  10301  10302  10303
1      39     44     40     41     49     42     44     47     46     46
1      27     35     28     33     31     31     37     32     34     34
1     173    169    166    161    158    169    137     41     10     24
1      31     28     34     32     35     34     35     35     37     39
1      27     34     35     34     36     34     39     35     37     38
..    ...    ...    ...    ...    ...    ...    ...    ...    ...    ...
40     39     34     33     37     34     38     41     33     37     40
40     87     86     87     92     88     85     91     85     90     84
40     91     92     93     90     90     92     89     93     93     93
40     40     34     38     37     32     41     40     36     39     40
40     39     34     32     30     38     27     36     36     35     34

[280 rows x 10304 columns]
```

Fig. 04: Train Data 7-3 Data Split

Test Data

	0	1	2	3	4	5	6	7	8	9	...
1	39	44	53	37	61	48	61	45	35	40	...
1	43	50	41	58	78	83	67	48	44	46	...
1	42	41	44	46	48	39	37	37	33	37	...
2	30	37	35	33	35	34	36	36	39	34	...
2	34	35	35	40	36	39	38	34	37	39	...
..	...	...	...	...	...	...	...	...	...	...	...
39	97	94	92	93	90	93	89	89	98	92	...
39	86	90	87	90	91	88	88	90	87	91	...
40	131	125	126	131	125	129	125	127	127	127	...
40	123	121	126	122	127	127	123	124	123	127	...
40	119	120	120	118	120	121	121	116	120	121	...

	10294	10295	10296	10297	10298	10299	10300	10301	10302	10303
1	23	30	36	32	28	32	31	29	26	29
1	167	164	164	162	159	156	155	158	153	169
1	42	33	39	39	40	41	43	42	43	41
2	27	54	47	44	37	29	25	26	27	32
2	26	27	23	29	26	27	26	31	24	24
..	...	...	...	...	...	...	...	...	...	...
39	67	38	46	121	115	109	129	144	143	129
39	132	132	127	131	139	139	137	127	124	126
40	91	89	90	90	93	88	93	89	93	91
40	29	47	34	36	42	34	39	40	35	42
40	63	88	87	97	101	91	93	89	94	85

[120 rows x 10304 columns]

Fig. 05: Test Data 7-3 Data Split

3. Perform the classification using Principal Component Analysis (PCA):

   A. Mean calculation

      Forming a vector with entities equivalent to the mean of each feature.

   B. Data Centring

      Shifting all the training samples by subtracting the previously calculated mean vector to centre the data around the mean of each feature (pixel value).

   C. Covariance Matrix Calculation

      Multiplying the transpose of the data matrix by the data matrix and dividing each element of the produced matrix by the number of samples (200 in case of 5-5 division and 280 in case of 7-3 division).

   D. Eigenvalues and Eigenvectors Computation

      Finding both eigenvalues and eigenvectors for the previously calculated covariance matrix.

   E. Fraction of Total Variance Calculation

      Eigenvalues are ordered in a descending order and the eigenvectors are in an order such that each eigenvector is in the place corresponding to its eigenvalue.

      Number of needed eigen vectors is calculated by accumulative summation of eigenvalues divided by the overall sum of all eigen values which gives a value between 0 and 1 that corresponds to the amount of variance that can be accepted for the model.

In this project this amount of variance is one of these 4 values: 0.8, 0.85, 0.9, and 0.95.

F. Eigenvectors Reduction

According to each one of the previously mentioned 4 values, an index is returned referring to the minimum number of eigenvectors needed to achieve this goal.

G. Projection of Data Matrix

This is done by multiplying the reduced eigenvectors' matrix transposed by the centred data matrix.

4. Use K Nearest Neighbours (KNN) classifier to produce the trained model:
   A. Perform the previous steps with each of the given values for the variable K: 1, 3,5, and 7.

      These values determine how many points is considered by the classification to take a decision and predict the corresponding class for a given image.

      This method chooses the most common class of the K points and assign it to the sample given. In case a tie occurred, the classifier chooses the first class to appear from these tied classes.

   B. Show the accuracy of prediction of test set samples with all possible combinations of K and α values.

      Note:

      For K = 7, if the data set was 5-5 divided, the accuracy will decrease as a tie will mostly occur which will probably cause some prediction errors. However, the accuracy will increase in case of 7-3 division.

Graphs

```
for Alpha =  0.8  : R =  16

Acc at K = 1: 93.5 %
Acc at K = 3: 86.0 %
Acc at K = 5: 84.0 %
Acc at K = 7: 77.5 %
```

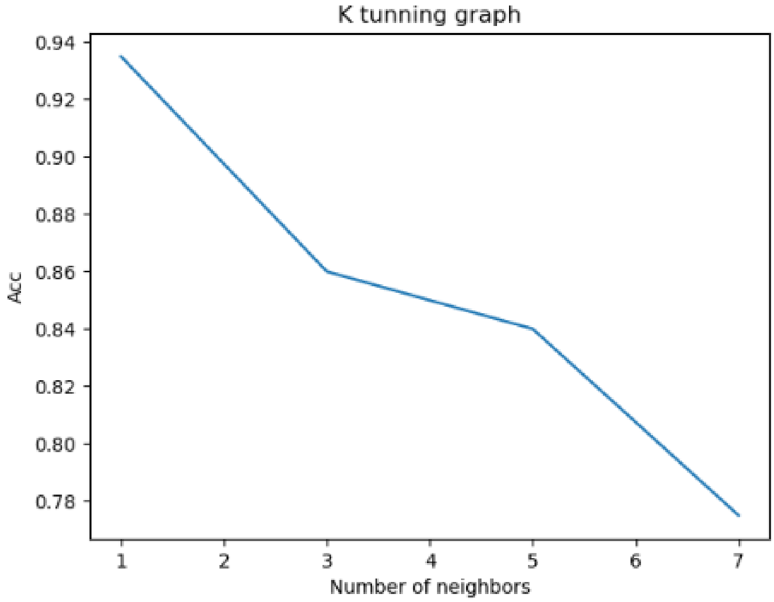

Fig. 06: Accuracy for α = 0.8 and 5-5 Data Split

```
for Alpha =  0.85  : R =  27

Acc at K = 1: 94.0 %
Acc at K = 3: 89.5 %
Acc at K = 5: 83.5 %
Acc at K = 7: 80.5 %
```

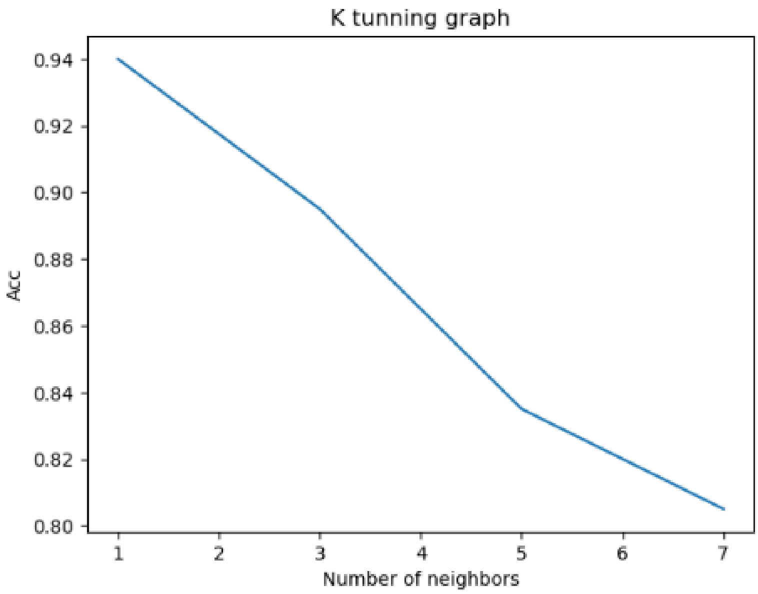

Fig. 07: Accuracy for α = 0.85 and 5-5 Data Split

```
for Alpha =  0.9  : R =  48

Acc at K = 1: 95.0 %
Acc at K = 3: 89.5 %
Acc at K = 5: 86.5 %
Acc at K = 7: 79.0 %
```

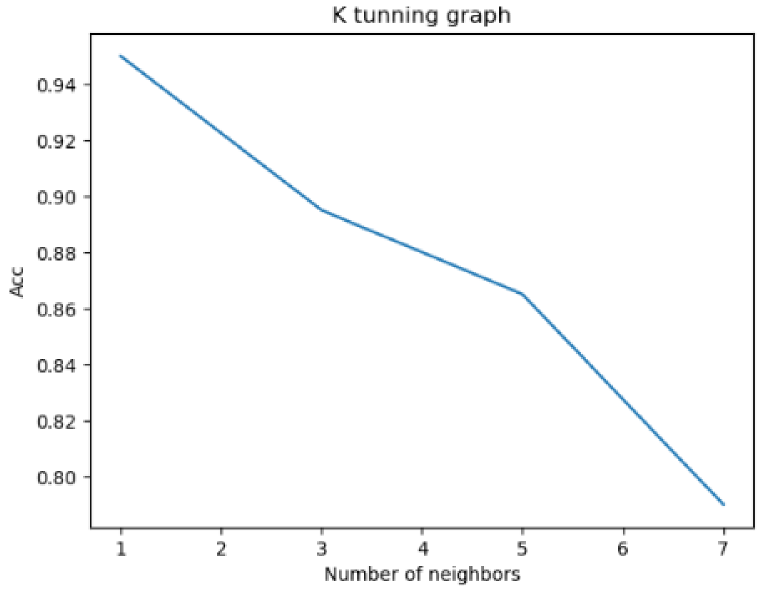

Fig. 08: Accuracy for α = 0.9 and 5-5 Data Split

```
for Alpha =  0.95  : R =  90

Acc at K = 1: 94.5 %
Acc at K = 3: 89.5 %
Acc at K = 5: 85.5 %
Acc at K = 7: 75.5 %
```

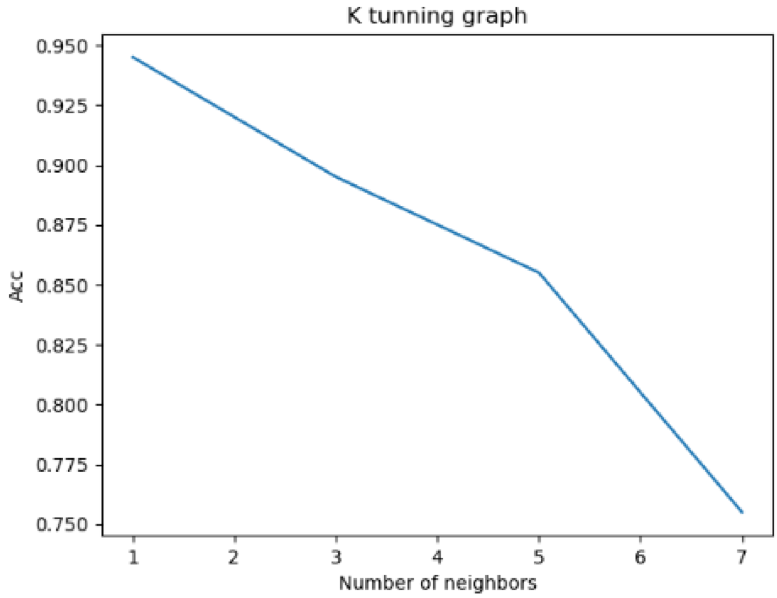

Fig. 09: Accuracy for α = 0.95 and 5-5 Data Split

```
for Alpha =  0.8  : R =  18

Acc at K = 1: 97.5 %
Acc at K = 3: 92.5 %
Acc at K = 5: 88.33333333333333 %
Acc at K = 7: 86.66666666666667 %
```

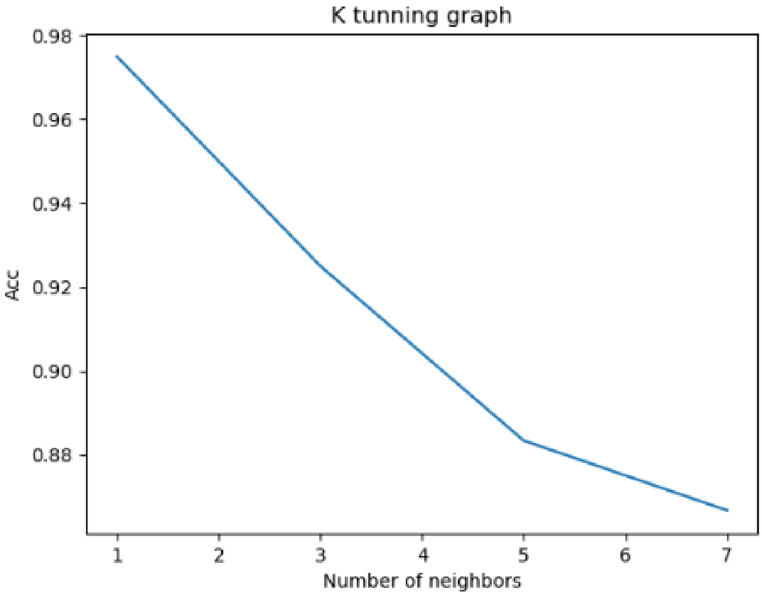

Fig. 10: Accuracy for α = 0.8 and 7-3 Data Split

```
for Alpha =  0.85  : R =  30

Acc at K = 1: 98.33333333333333 %
Acc at K = 3: 91.66666666666666 %
Acc at K = 5: 90.0 %
Acc at K = 7: 90.0 %
```

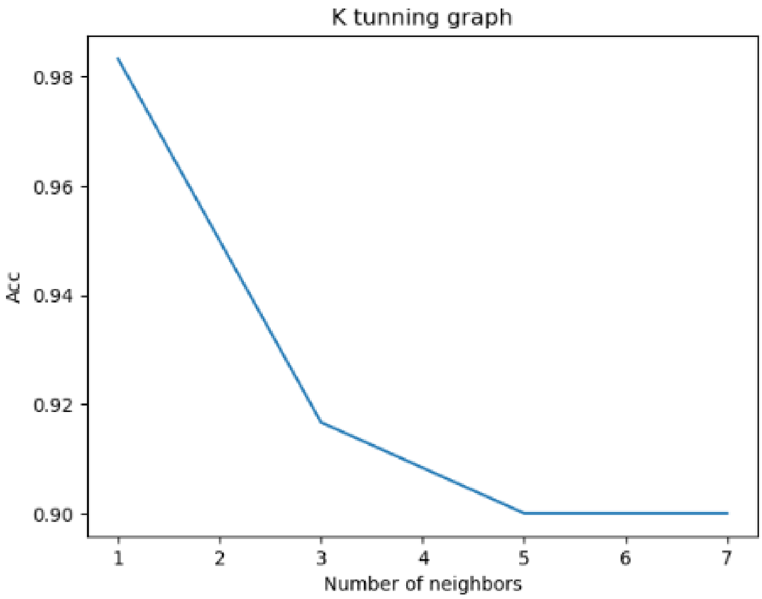

Fig. 11: Accuracy for α = 0.85 and 7-3 Data Split

```
for Alpha =  0.9  : R =  55

Acc at K = 1: 98.33333333333333 %
Acc at K = 3: 96.66666666666667 %
Acc at K = 5: 91.66666666666666 %
Acc at K = 7: 90.0 %
```

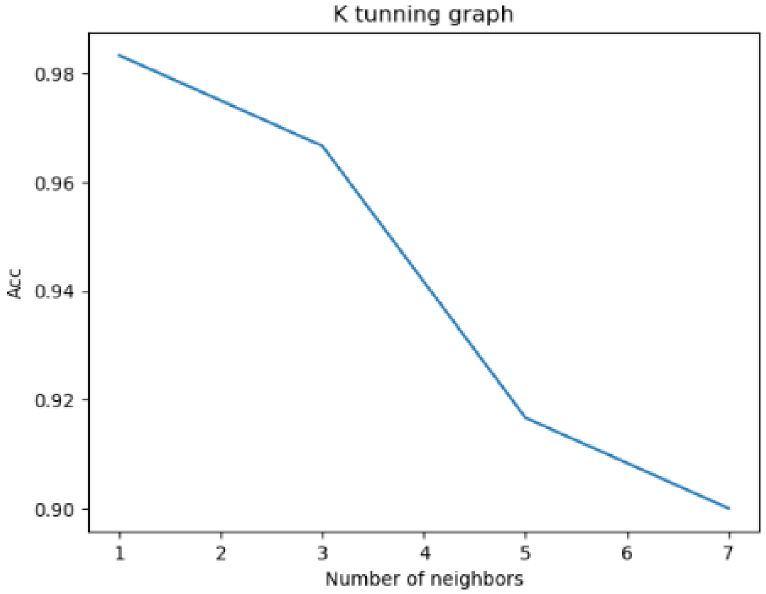

Fig. 12: Accuracy for α = 0.9 and 7-3 Data Split

```
for Alpha =  0.95  : R =  111

Acc at K = 1: 98.33333333333333 %
Acc at K = 3: 96.66666666666667 %
Acc at K = 5: 93.33333333333333 %
Acc at K = 7: 90.0 %
```

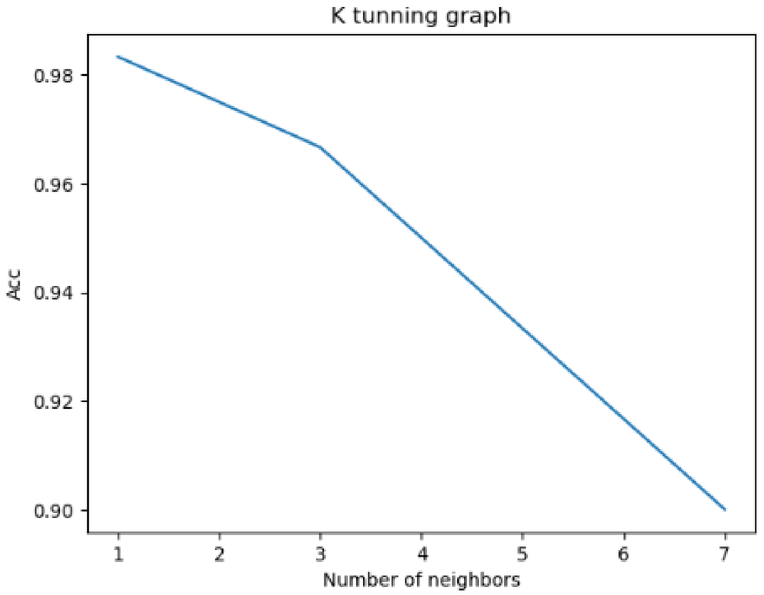

Fig. 13: Accuracy for α = 0.95 and 7-3 Data Split