**Problem** : Principal Component Analysis

1. Perform PCA on the training set. Plot the mean face and the first four eigenfaces.

|  |  |  |  |
| --- | --- | --- | --- |
| Mean face | G:\hw1\mean.png |  |  |
| G:\hw1\pca1.png | G:\hw1\pca2.png | G:\hw1\pca3.png | G:\hw1\pca4.png |
| Eigenface1 | Eigenface2 | Eigenface3 | Eigenface4 |

1. Take person2image1, and project it onto the PCA eigenspace you obtained above. Reconstruct this image using the first n = 3; 50; 170; 240; 345 eigenfaces. Plot the five reconstructed images.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| G:\hw1\3.png | G:\hw1\50.png | G:\hw1\170.png | G:\hw1\240.png | G:\hw1\345.png |
| N=3 | N=50 | N=170 | N=240 | N=345 |

1. For each of the four images you obtained in 2., compute the mean squared error (MSE) between the reconstructed image and the original image. Record the corresponding MSE values in your report.

N = 3, MSE= 746.7993871598167

N = 50, MSE= 238.62252486497164

N = 170, MSE= 44.79377905553697

N = 240, MSE= 12.480908162325504

N = 345, MSE= 0.21543397076006804

1. Now, apply the k-nearest neighbors algorithm to classify the testing set images. First, you will need to determine the best k and n values by 3-fold cross-validation. For simplicity, the choices for such hyper parameters are k = {1, 3, 5} and n = {3, 50, 170}. Show the cross validation results and explain your choice for (k, n).

k= 1 , n= 3 , accuracy= 0.6749999999999999

k= 1 , n= 50 , accuracy= 0.9638888888888889

k= 1 , n= 170 , accuracy= 0.9527777777777778

k= 3 , n= 3 , accuracy= 0.5861111111111111

k= 3 , n= 50 , accuracy= 0.8638888888888889

k= 3 , n= 170 , accuracy= 0.8666666666666667

k= 5 , n= 3 , accuracy= 0.5583333333333333

k= 5 , n= 50 , accuracy= 0.8000000000000002

k= 5 , n= 170 , accuracy= 0.8083333333333332

The cross-validation result with k=1 n=50 parameter have the highest accuracy so choose k=1 n=50.

1. Use your hyperparameter choice in 4. and report the recognition rate of the testing set.

the recognition rate of the testing set is 0.925

Ref: [1] <https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>

[2] <https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>

[3] https://blog.csdn.net/HLBoy\_happy/article/details/77146012