Neural Networks
Group 04
Andrea Jemmett (2573223)
Bas van der Drift (1707507)

Assignment 5 - Principal Component Analysis

The main Matlab script for this assignment can be found attached as ‘ass5.m’.

Netlab’s *pca* function was used to apply Principal Component Analysis (PCA) to the image dataset supplied in the pics.mat file. In order to be able to analyse what the influence of the encoding and decoding is on the image for a varying number of components, a vector was created containing the different numbers of components to test (1 to 60, 61 to 199 with step size 5, 200 to 2000 with step size 100, 2576).

The distance between the original image and it’s encoded and decoded form was measured using the function explained below. It computes the correlation coefficient between two images in the following way:

$$r = \frac{\sum_m \sum_n (A_{mn} - \overline{A})(B_{mn} - \overline{B})}{\sqrt{\left(\sum_m \sum_n (A_{mn} - \overline{A})^2\right)\left(\sum_m \sum_n (B_{mn} - \overline{B})^2\right)}}$$

so that the higher *r* corresponds to more similar images (*r* = 1 means that are the same). In this formula, A and B are the images being compared and m and n are the x and y coordinates of the pixels in the image. Summarizing: it computes the difference between the mean image intensity and the intensity value for each pixel in the image. For our purposes this is a very useful performance metric, since it gives us a good general idea about how much information was lost in the encoding and decoding.

As we expected, a higher number of principal components gives an increased similarity between the original and processed image. Figure 1 displays the relationship between the number of components used and the correlation coefficient. The mean correlation coefficient reaches 0.902 at when using 76 components, which indicates this might be the optimal number of principal components to use when looking for a combination of dimensionality reduction and feature representation. Depending on the needs of the particular application a choice can be made to use more or less components, resulting in stronger dimensionality reduction or stronger data integrity.

The mean correlation approaches 1 when extending the number of principal components to the maximum of 2576 with a logarithmic behaviour, as shown in figure 1 by means of a semi-logarithmic plot:

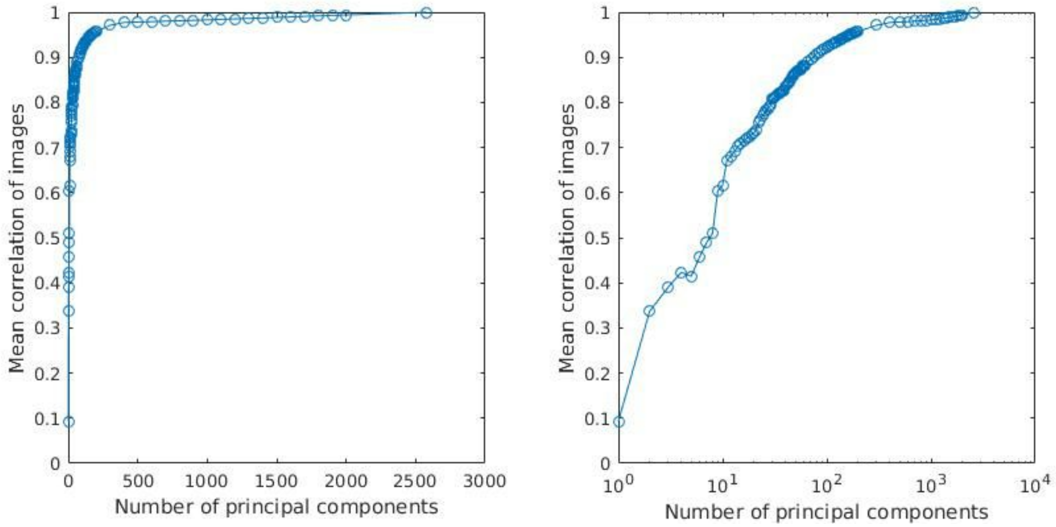
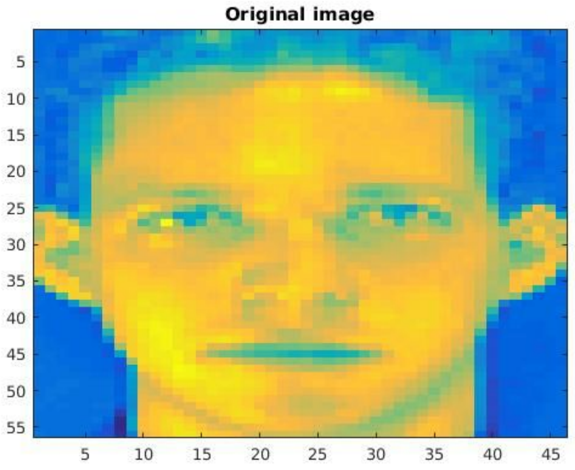
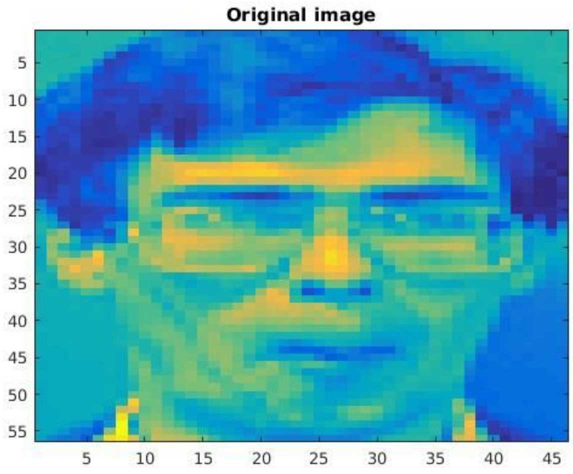


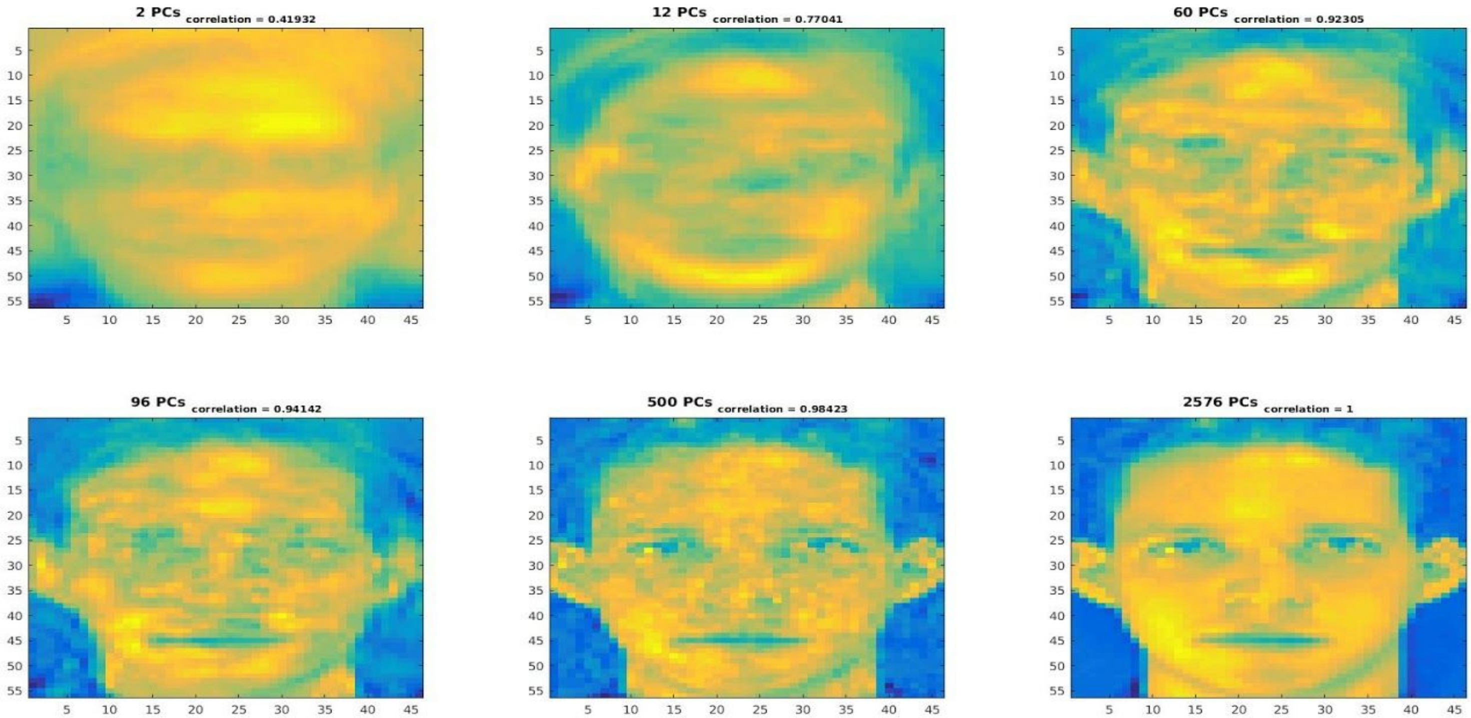
Figure 1: Mean correlation approaches and reaches 1 when moving toward 2576 principal components.

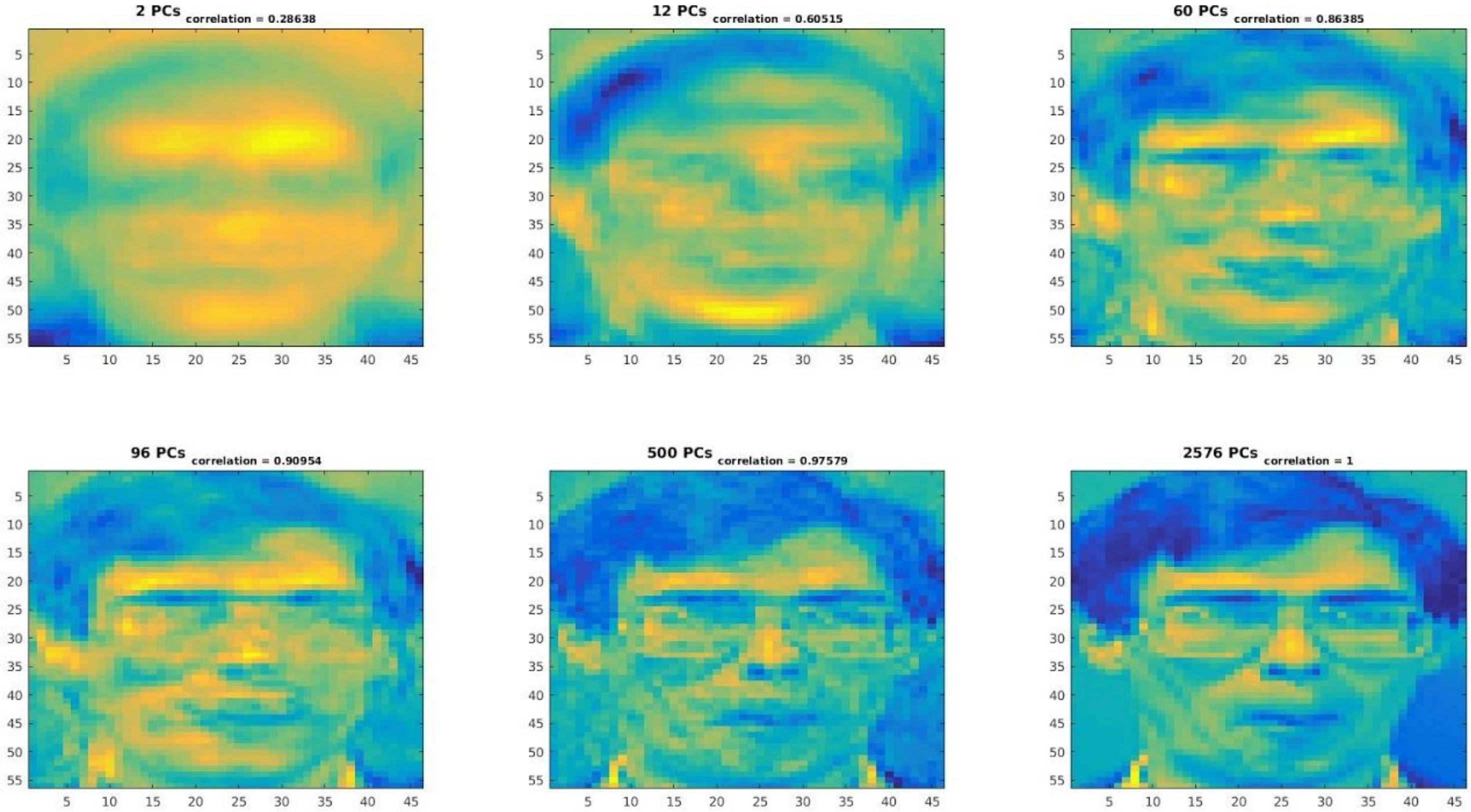
Since the number of pixels in an image is 2576, using 2576 principal components means no information is lost and the encoded and decoded images are exactly the same as the original images. For all practical purposes this is useless, but it nicely illustrates the idea of what dimensionality reduction through PCA means for the dataset.

Below the same image is showed using different numbers of principal components to further illustrate the idea of PCA: The more components are used, the more the image resembles the original. A PCA with 2576 components would look exactly like the original.






2 PCs correlation = 0.41932
12 PCs correlation = 0.77041
60 PCs correlation = 0.92305
96 PCs correlation = 0.94142
500 PCs correlation = 0.98423
2576 PCs correlation = 1
5
10
15
20
25
30
35
40
45
50
55
5
10
15
20
25
30
35
40
45


2 PCs correlation = 0.28638
12 PCs correlation = 0.60515
60 PCs correlation = 0.86385
96 PCs correlation = 0.90954
500 PCs correlation = 0.97579
2576 PCs correlation = 1
5
10
15
20
25
30
35
40
45
50
55
5
10
15
20
25
30
35
40
45

In the following table the mean correlation between all original and encoded and decoded images is shown.

Number of principal components	2	12	60	96	500	2576
Mean correlation	0.337	0.680	0.882	0.920	0.978	1

The challenge when using PCA is to select the right number of components to use for further analysis or learning, which depends on the dimensionality of the original data and the purpose of the further analysis or learning. We imagine in practically all cases one would try to find a balance between dimensionality reduction and data integrity. For example for a task like classify images as containing glasses or not, probably a number of components less than 100 would not be enough to capture meaningful features; to detect instead if a picture contains a human face or not a number of PCs between 12 and 60 may suffice.