University of Nevada, Reno
Department of Computer Science & Engineering

CS 679: Programming Assignment 3

Instructor: Dr. George Bebis

Submitted by: Tyler Becker, Mayamin Hamid Raha

Date of Submission: 04/25/2023

Contribution: Model training for experiments A-D, all debugging of all sections was implemented by Tyler Becker, and the implementation for the corresponding tests and experiments was done by Mayamin Hamid Raha.

Theory

In this section the underlying theories behind this facial recognition classifier will be discussed.

Curse of Dimensionality:

The main motivations for this experiment are to explore how well our classifiers can do in problems which have high dimensionality. In these problems the classifiers experience what is referred to as the curse of dimensionality. Essentially, this curse can be explained as follows:

Increasing the number of calculated features does not necessarily increase the accuracy of the classifier, and the amount of data required to estimate the parameters of the classifier is exponentially related to the dimensionality of the feature set.

However, there are many problems which contain feature sets of high dimensionality. When classifying images each pixel contains information about the image and is therefore a feature. Because of this it becomes necessary for us to reduce the dimensionality of our feature set while still retaining the information that the features contained.

Principal Components Analysis (PCA):

The method we will use to reduce the dimensionality of our features is called Principal Components Analysis (PCA). The main concept of this method is to approximate the distribution of the data in a subspace which has significantly less dimensions than the feature spaces and minimizes the loss of information. In order to do this we first calculate the covariance matrix through the following method (see Equation 1-3).

$$\phi_i = x_i - \bar{x} \qquad \text{..... (1)}$$

$$A = \left[\phi_1 \,|\, \phi_2 | \dots | \phi_M \right] \qquad \text{..... (2)}$$

$$\Sigma_x = \frac{1}{M} A A^T \qquad \text{..... (3)}$$

If we have features represented by x where x $\in R^D$, then PCA tries to approximate x (see Equation 4). in a subspace of R^D and finds a new set of orthogonal basis vectors $<u_1, u_2, \dots, u_k>$, $u_i \in R^D$

$$\hat{x} = \sum_{j=1}^{K} y_i u_i \quad \text{where} \quad y_i = \frac{x^T u_i}{u_i^T u_i} \qquad \text{...... (4)}$$

The approximation of x is done by PCA such that the distance between the projected image and the original image is the minimum. This makes sure that information loss is minimized. Then it becomes possible to reconstruct images from these coefficients if necessary. For calculation of projection coefficient Equation 4 is used where Equation 5 and 6 is followed.

$$u_i^T u_i = 1 \; if \; i = j \qquad \text{........(5)}$$

$$u_i^T u_i = 0 \;\; \text{otherwise} \qquad \text{........(6)}$$

It is important to note that the eigenvectors we keep are the ones which represent the highest variance in the data. These eigenvectors correspond to the eigenvalues of greatest magnitude and therefore the “largest” eigenvectors are what the data is projected onto.

Eigen Faces :

Eigenface is a set of eigenvectors that are used for facial recognition [1] . Eigenvectors are derived from covariance matrix calculated from distribution of data in high dimensional vector space. The eigenvectors that are also basis vectors can be used to represent original training images allowing huge reduction in dimension .i.e. only a smaller set of training images can be used to represent original images. One property that makes eigenvectors unique is that they don’t change orientation when a transformation is applied (change of coordinate system or space). Use of eigenfaces for facial recognition is much faster and computationally efficient than facial recognition using pixel level classification.

For a given picture its mean-subtracted version can be projected onto eigenface pictures using a vector dot product. The result of this dot product shows how close the picture is to the eigenface picture. For Facial Recognition using eigenfaces, the first step is to calculate k largest eigenvectors from the training data provided that the data is centered and of the same size. This is done using principal component analysis. An average face also needs to be computed from all images in the dataset which will be used to center the test images (Equation 1). After this step, the test image needs to be projected onto the eigenspace of every train image (Equation 4,7).

$$y_i = \Phi^T u_i \qquad \text{.........(7)}$$

For recognition tasks a metric known as distance in face space (difs) needs to be calculated. This is done by comparing the projection coefficients of each test image from every train image.

$$\text{p} = arg\ min_i \, || \, \Omega - \Omega_i \, || \quad i = 1, 2..... M \qquad \text{..... (8)}$$

where M is the number of training data.

In order to find the closest distance of test projection coefficients from every train projection coefficient, Mahalanobis distance is used. (see Equation 9).

$$e_r = min_i \frac{1}{\lambda_j} \sum_{j=1}^{K} (y_j - y_j^i)^2 \qquad \text{.........(9)}$$

Finally for recognizing faces of people, the training image whose distance in face space is the lowest from the test image is selected as the face recognized. For application fields such as intruder detection a threshold (lets say T_r) is imposed in the distance in face space, i.e. $e_r < T_r$.

Cumulative Match Characteristics Curve (CMC):

CMC curve is used for measuring performance of identification systems. It takes into account the ranking capabilities of the identification system. CMC curve compares rank versus identification rate.

Results and Discussion

In this section the experiments for this assignment will be described and then their results will be discussed.

Classification Experiment

(a.I)

For this experiment we simply trained the classifier on the High resolution data from the provided data set. The data was loaded in and used to calculate the covariance matrix, and from there the eigenvectors were estimated. Figure 1 shows the average face from the data, which is fairly standard.

Interestingly the eigenfaces, or image projections of the eigenvectors we estimated, show different regions of interest where the information is kept for each eigenvector. In Figure 2 the top ten eigenfaces are shown where these regions are particularly visible, for instance some of the images show lighter around the eyes or the upper lip etc. This is a visual representation of the features that are then being used for classification. Figure 3 then shows the last ten eigenfaces or the eigenfaces that correspond to the lowest values. The images are noisy which shows that the lowest eigenvectors contain very little information about the data.

Fig 1: High Resolution Average Face

Fig 2: Eigen faces corresponding to 10 largest Eigenvalues for High Resolution Image Dataset

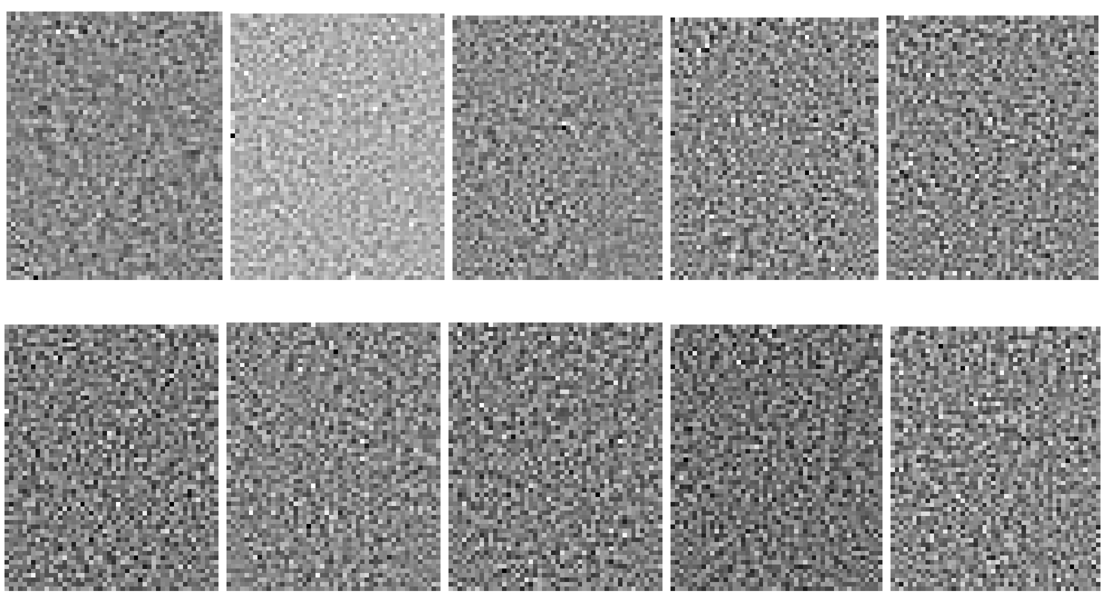
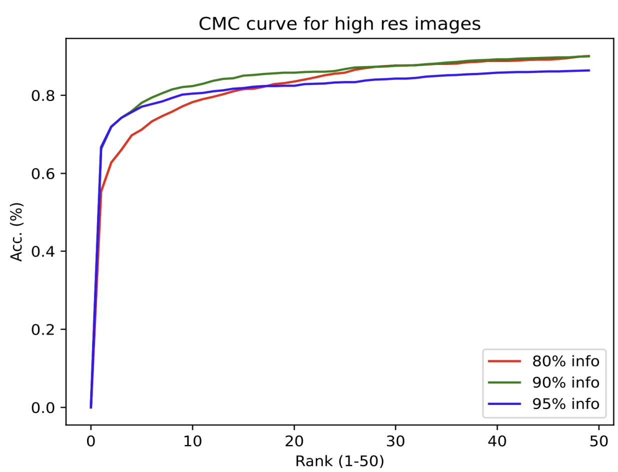


Fig 3: Eigen faces corresponding to 10 largest Eigenvalues for High Resolution Image Dataset

(a.II, a.V)

We then performed classification of the images on the dataset and obtained CMC curves for high and low resolution. We found that the face recognition algorithm performs better on low resolution images than high resolution images in Figure 4. This in general could be due to the Curse of Dimensionality. The images in the higher resolution data set have significantly more features and therefore it is not only possible that we did not have enough data to compute the features, but also that the features became less expressive of the contents of the image as well. Also of note in the high resolution, the highest amount of data kept had the lowest performance overall although it grew the fastest with respect to the rank. This implies that only the first few eigenvectors contain the bulk of the information as we assumed prior to conducting the experiment, and that to a certain extent the extra dimensionality helped to make the dataset more separable. For the low resolution data our model performed as expected and when we increased the retained information, and the rank our accuracy grew with the best results being obtained when we kept the most amount of the information.



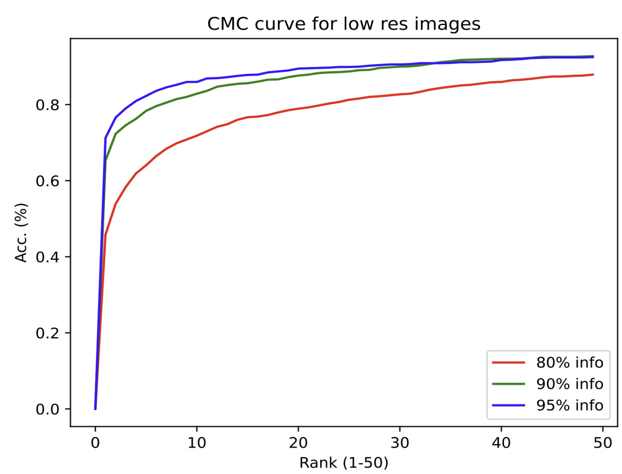


Fig 4 : CMC plot for High Resolution Images and Low Resolution images

(a.III)

The following figures show images for each amount of information retained, that were classified correctly with a rank of one, and were not classified correctly with rank of one for each amount of information kept respectively.

Fig 5: 3 High Resolution images that are correctly classified (for 80% information in data)

Fig 6: 3 High Resolution images that were incorrectly classified (for 80% information in data)

Fig 7: 3 High Resolution images that are correctly classified (for 90% information in data)

Fig 8: 3 High Resolution images that are incorrectly classified (for 90% information in data)

Fig 9: 3 High Resolution images that are correctly classified (for 95% information in data)

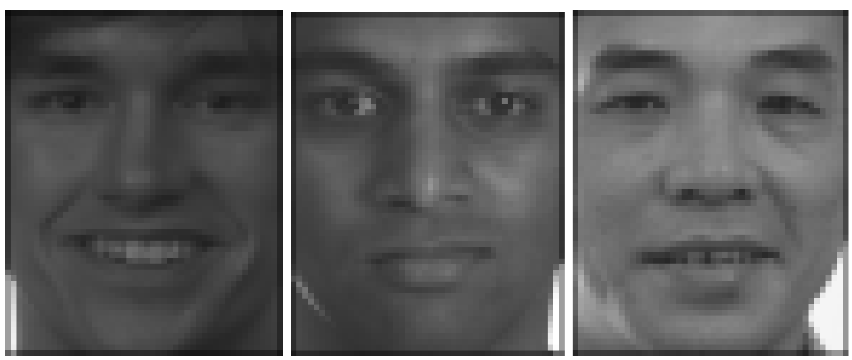


Fig 10: 3 High Resolution images that are incorrectly classified (for 95% information in data)

(c) For low resolution dataset the experiment a has been repeated and we report the following (see Figure 11 - 17) :

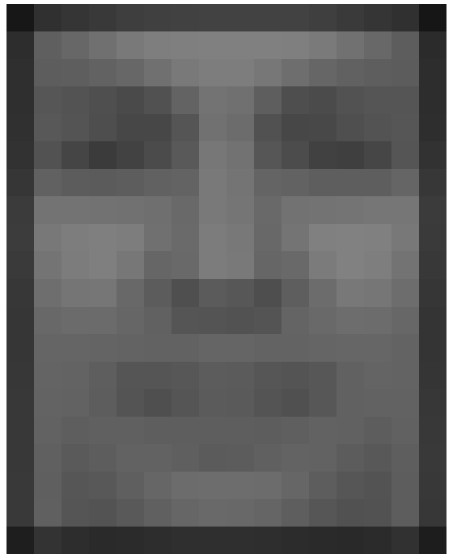


Fig 11: Average face Low Resolution

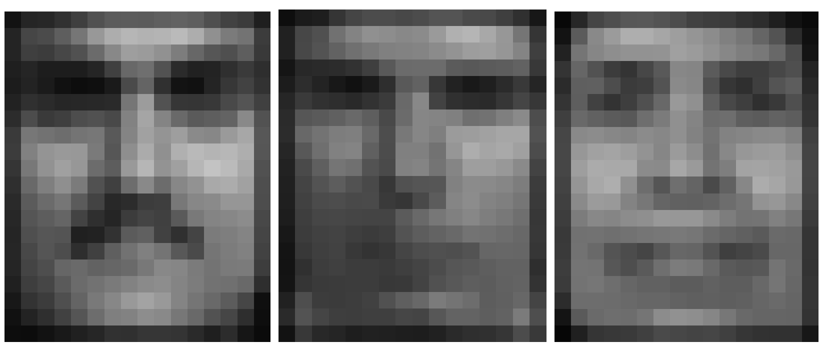


Fig 12: 3 LowResolution images that are correctly classified (for 80% information in data)

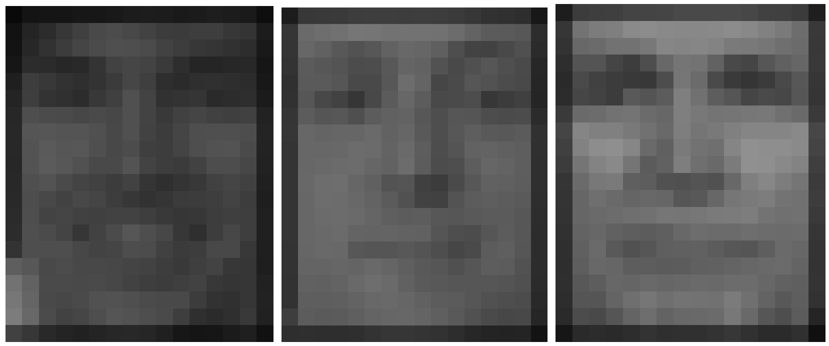


Fig 13: 3 LowResolution images that are incorrectly classified (for 80% information in data)

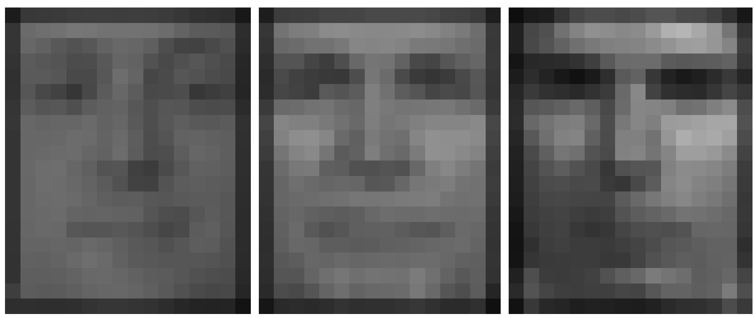


Fig 14: 3 LowResolution images that are correctly classified (for 90% information in data)

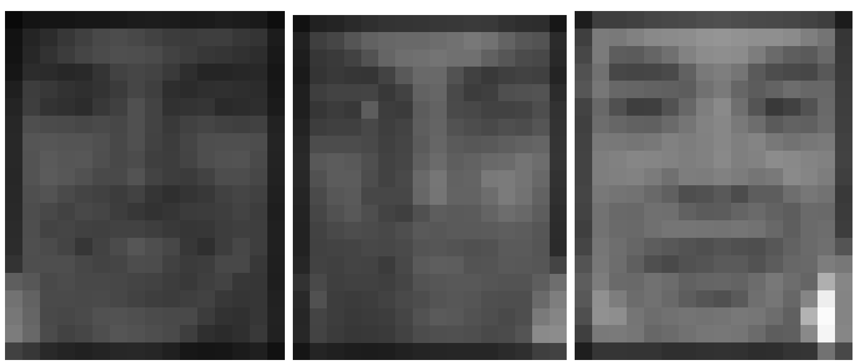


Fig 15: 3 LowResolution images that are incorrectly classified (for 90% information in data)

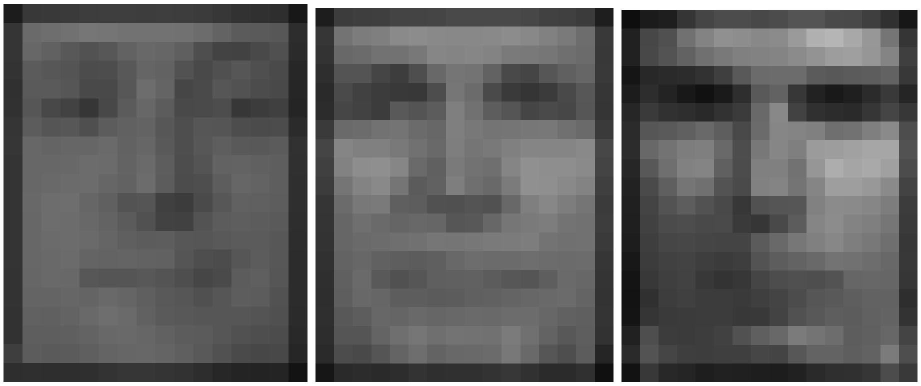


Fig 16: 3 LowResolution images that are correctly classified (for 95% information in data)

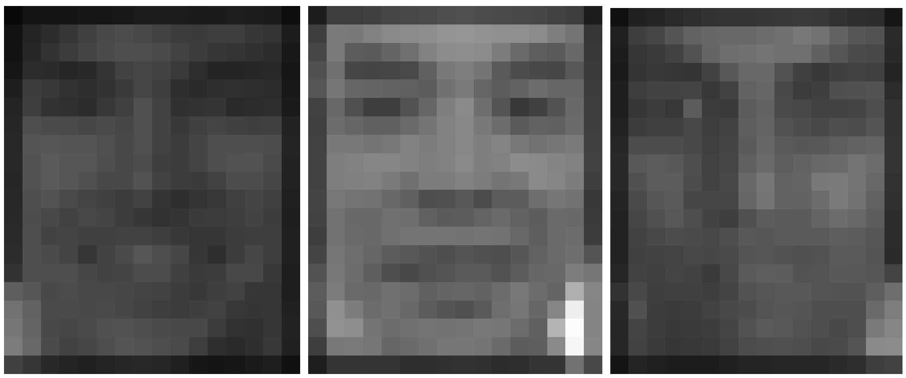


Fig 17: 3 LowResolution images that are incorrectly classified (for 95% information in data)

Intruder Detection Experiment

In this experiment we removed images corresponding to labeled individuals that do appear in the test set in order to simulate intruder detection. For this the first 50 labeled individuals were removed from both the low resolution and high resolution data sets and the models were retrained. From there we tested different thresholds on the data and created ROC curves from the results of classification. In this experiment we primarily cared about the amount of False Positives we received based on the different thresholds and therefore we plotted false positives against true positives in terms of proportions to the amount of actual intruders, and the amount of non intruders.

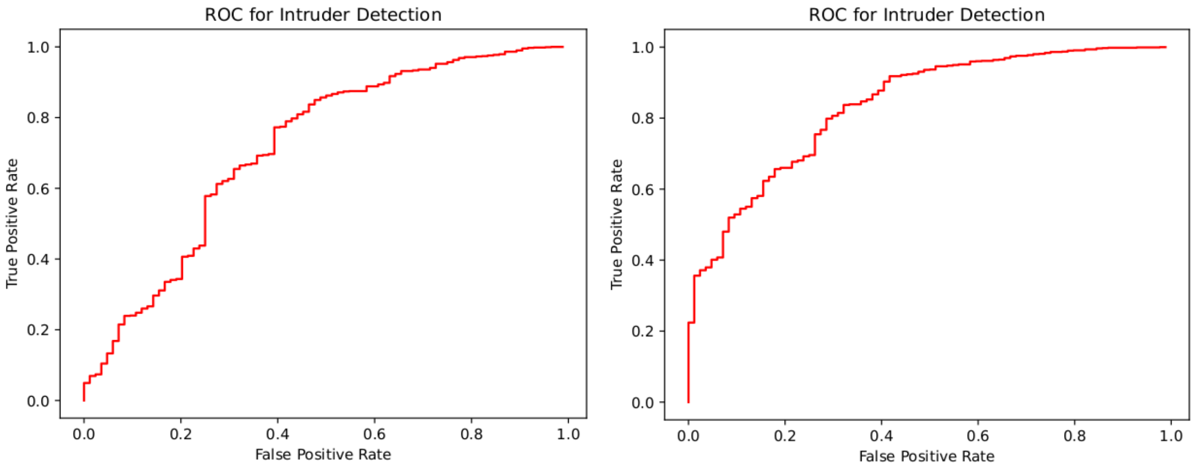


Fig 18: ROC curve results for intruder detection for both the low resolution and high resolution data set. The results for high resolution are shown in the left graph, and low resolution on the right.

Figure 18 shows both ROC curves yielded from this experiment. Again we found better results from the low resolution data which is shown in the fact that the low resolution data comes closer to the top left corner of the graph and therefore has a higher area under the curve. This is again due to the curse of dimensionality.

Conclusion

Overall, the classifier performed fairly well on both data sets, however, we did get better results when testing on lower resolution data across all experiments as opposed to higher resolution data. This implies that the design of the classifiers will not scale well into higher dimensional data. This could be due to simply the curse of dimensionality, however, it also seems that the problem is simpler in lower resolution when the pictures are taken from the same distance. There are less features defining the parts of a person's face and therefore the faces are in general more homogenous than when they are displayed in a higher resolution. This meant that in the first experiment the solutions performed worse at lower ranks, however, performed better overall at higher ranks.

Reference

[1] Turk, Matthew A., and Alex P. Pentland. "Face recognition using eigenfaces." *Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition*. IEEE Computer Society, 1991.