Pattern Recognition

Assignment 3

CS 479/679 Pattern Recognition (Spring 23)
Programming Assignment 3

Shankar Poudel & Aminul Huq
Contribution: Equal contribution in both coding as well as report writing.

Computer Science and Engineering Department
University of Nevada, Reno

April 24, 2023

1 Part 1: Theory

Dimensionality Reduction

Training a model to perform some sort of classification, it is sometimes noticed that the model is not performing well. A common assumption in this kind of scenario is to increase the number of features of the data so the model can have more information and perform well. However, this approach is not always good because increasing the number of features causes a problem called the curse of dimensionality. It occurs when we have too many features of the data but not enough samples of data. So there are scenarios where the it is nice to have a lower number of features. However, deciding which features are the most important ones i.e. which have the most information or have the most discriminatory features is a difficult thing to decide. Dimensionality reduction helps in such scenarios by performing feature extraction on the training data. Based on different objectives like minimizing information loss or maximizing discriminatory features there are a few dimensionality reduction approaches. Principal component analysis (PCA) is a dimensionality reduction method which can be used as a feature extraction technique that focuses on reducing the information loss.

Principal Component Analysis(PCA)

In PCA the goal is to minimize the information loss. In order to attain this goal, this approach tries to preserve information where it has the most variance of the data. It tries to come up with a set of vectors upon which the input data is projected. PCA calculates the eigenvalues and corresponding eigenvectors of the data and uses the largest pair of eigenvalues and eigenvectors. The eigenvalues correspond to the variance of the data along the eigenvector directions. The overall steps to calculate PCA are discussed below:

Suppose we have a total of M number data and d number of features and the inputs can be defined as $x_1, x_2, ...x_d$. At first we will be computing the mean of the input data and then subtract the mean from each data point. This is done so that all the data are centered at zero which helps PCA.

$$\bar{x} = \frac{1}{M}\Sigma_{i=1}^{M} x_i \tag{1}$$

$$\phi_i = x_i - \bar{x} \tag{2}$$

Since the data is in standardize format now, we will be calculating the sample covariance matrix Σ_x of the input data

$$\Sigma_x = \frac{1}{M}\Sigma_{i=1}^{M}(x_i - \bar{x})(x_i - \bar{x})^T \tag{3}$$

$$\Sigma_x = \frac{1}{M}\Sigma_{i=1}^{M}\phi_i\phi_i^T \tag{4}$$

$$\Sigma_x = \frac{1}{M}AA^T \tag{5}$$

In equation (5), $A = [\phi_1, \phi_2, ..., \phi_M]$ which means the columns of A are the ϕ_i. In the next step we will be computing the eigenvalues and eigenvectors of the covariance matrix $\Sigma_x$and ordering the pair which has the most variance to the lowest ones.

$$\Sigma_x u_i = \lambda_i u_i \tag{6}$$

where the values of λ_i represent the eigenvalues and u_i represent the eigenvectors. Since Σ_x is a symmetric matrix then $u_1, u_2, ..., u_d$ will form an orthogonal basis in R^d. Hence we can represent any $x \epsilon R^d$ as:

$$\boldsymbol{x} - \hat{\boldsymbol{x}} = \Sigma_{i=1}^{d} y_i u_i = y_1 u_1 + y_2 u_2 + ... + y_d u_d. \tag{7}$$

Here, y_i are the eigencoefficients. We can now perform dimensionality reduction by approximating $\boldsymbol{x}$ by $\hat{\boldsymbol{x}}$ by using only the first K eigenvectors which correspond to the largest K eigenvalues where $K << d$. K is a parameter which helps us choose how much information we want to preserve in the data. Hence equation (7) can be written as,

$$\boldsymbol{x} - \hat{\boldsymbol{x}} = \Sigma_{i=1}^{K} y_i u_i = y_1 u_1 + y_2 u_2 + ... + y_K u_K. \tag{8}$$

In scenarios where we have a large dimensions of features of the data, performing eigenvalue and eigenvector calculation is computationally expensive. Suppose if we have M images of size $N \times N$ then each image can be represented as $N^2 \times 1$ 1D vector then the convariance matrix would be $N^2 \times N^2$. In order to avoid this computationally expensive task there is a trick to reduce the number of calculation. In that case the covariance matrix would become M. This can be done by changing equation (5) slightly from AA^T to $A^T A$. We can rewrite equation (6) as:

$$(A^T A) v_i = \mu_i v_i \tag{9}$$

Multiplying both sides with A we get:

$$A(A^T A) v_i = A \mu_i v_i \tag{10}$$

$$(AA^T)(Av_i) = \mu_i (Av_i) \tag{11}$$

Now if we try to format equation (10) to equation (6) then we will get,

$$\lambda_i = \mu_i \text{ and } u_i = Av_i \tag{12}$$

Using this above mentioned trick the number of computation comes down significantly. The M eigenvalues and eigenvectors of $A^T A$ correspond to the M largest eigenvalues and eigenvectors of AA^T.

Model Comparison

We are using two ways to compare the model, Cumulative Match Characteristic (CMC) curve and ROC graph.

a) Cumulative Match Characteristic (CMC) curve: CMC curve shows how good a model is predicting given the true level falls under first 'r' predictions. It shows a measure of 1:r identification system performance. A curve that reaches to the higher value of performance faster is regarded as better model.

b) Receiver Operating Characteristic (ROC) curve: A ROC curve is a graph showing the performance of a classification model at various thresholds. This curve is a plot of two parameters: True Positive Rate VS False Positive Rate. A model with higher area under ROC-curve (AUC) is regarded better than model with low AUC for given problem.

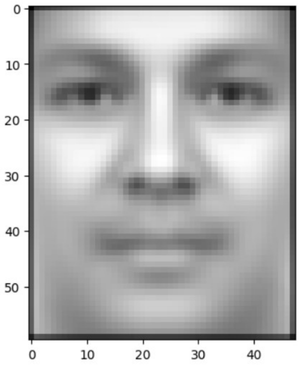
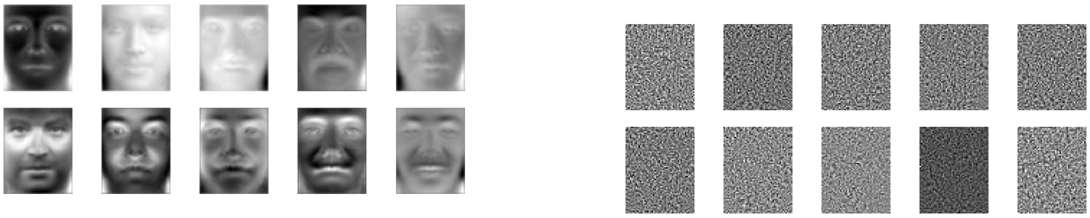


Figure 1: Average face of training images of Higher Resolution.



(a)

(b)

Figure 2: Eigenfaces for (a) largest 10 components and (b) smallest 10 components for training data of Higher resolution.

2 Part 2: Results and Discussions

2.1 Dataset Description

There are a total of 4 sets of data namely fa_h, fb_h, fa_l and fb_l. For training the PCA we are using the fa_h and fa_l data and for testing we are using fb_h and fb_l. 'h' and 'l' represent high and low dimensional data. For 'h' and 'l' we have images with the dimensions of 48 × 60 and 16 × 20. For all the following subsections that contain a in the title we used fa_h for training and fb_h for testing.

2.2 a.I

The average face (figure 1) shows the smooth face structure of a generic person. It takes the average of each person's photo from the training data and displays the average faces image.

Eigenfaces for the largest and smallest 10 eigenvalue components can be found in Figure2. From the largest 10 eigenfaces we get the eigenvalues which correspond to the components which varies the most in this training dataset. Generally, the first three components correspond to the illumination of the data. These eigenfaces have the most information of the overall data. However, the eigenfaces which correspond to the smallest 10 eigenvalues are not that important as they contain most noises. These eigenfaces varies the least amount.

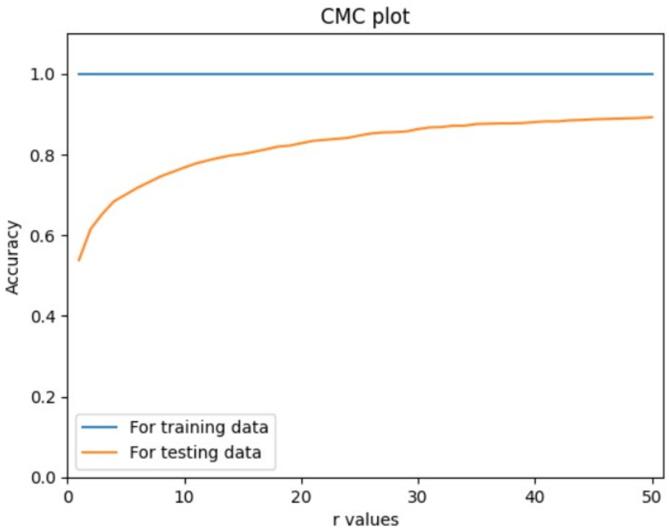


Figure 3: CMC curve for 80% of data information retention in Higher Resolution training data.

2.3 a.II

For this experiment we retrained 80% of information from the training data(fa_h) and used the testing data(fb_h) images as the query images and created a cumulative match characteristics curve(CMC). We tried to plot the top r matching samples in a cumulative manner which are able to correctly classify the unknown images. We used Mahalanobis distance as a measuring tool for distance mapping between the query samples and the training samples. We have set the value of r from 1 to 50 and compared the performance of both the training samples and testing samples. The x and y axis of the curve represent the values of r and accuracy metrics respectively.

From the Figure 3 we can see that on the training data with only 80% of information retention it can achieve 100% accuracy even when the value of r was 1 i.e. the first sample was the correct answer for all the training data. For test data we got less than 60% of accuracy for r=1. However, as the value of r increased the overall performance increased as well. The highest accuracy that was able to achieve was less than 90% and this was for r = 50. Here we see that performance of the model with low information i.e. model trained with information retention of 90%, does better than model with high information. This is mainly due to curse of dimentionality, and proves that on increasing the number of features doesnot always increase the perofrmance but after a certain point it causes to decrease too.

2.4 a.III

In this section, we are displaying the three samples which are correctly matched for r=1 meaning the samples which match with the training samples on the first try. In Figure 4a, the first column of images represent the query samples and the second column represent the images with the matched samples. From the figure we can see that even with change of illumination, face expression etc with 80% of information preserved these samples were

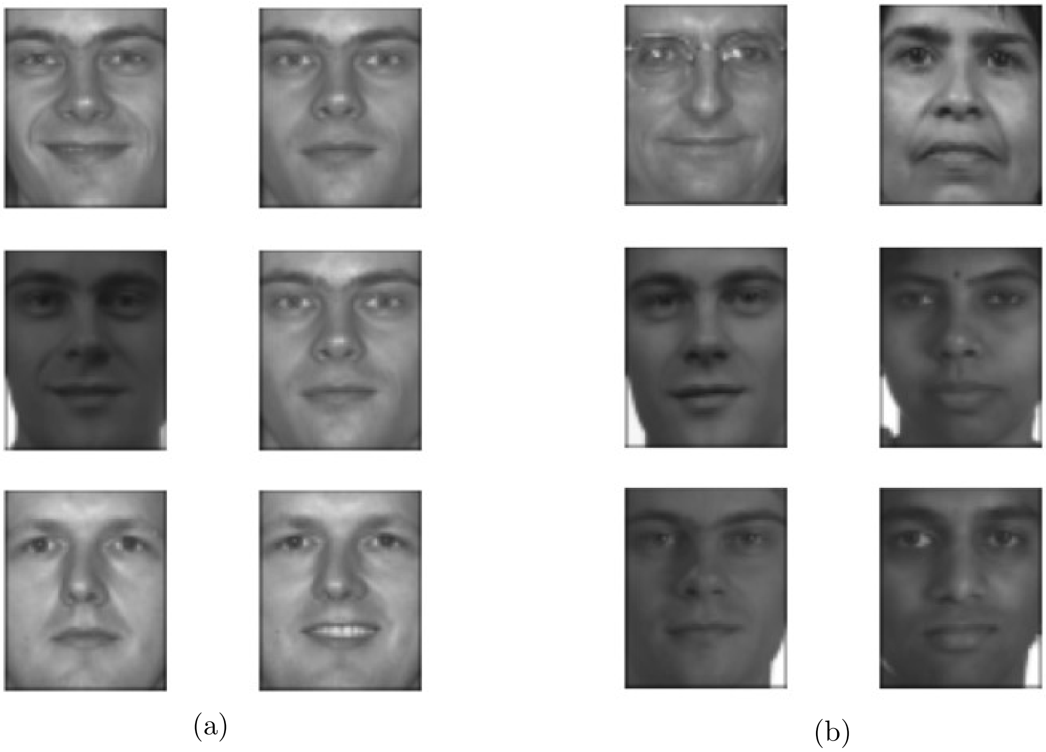


Figure 4: (a) Correctly matched samples and (b) Incorrectly matched samples when querying in data of higher Resolution.

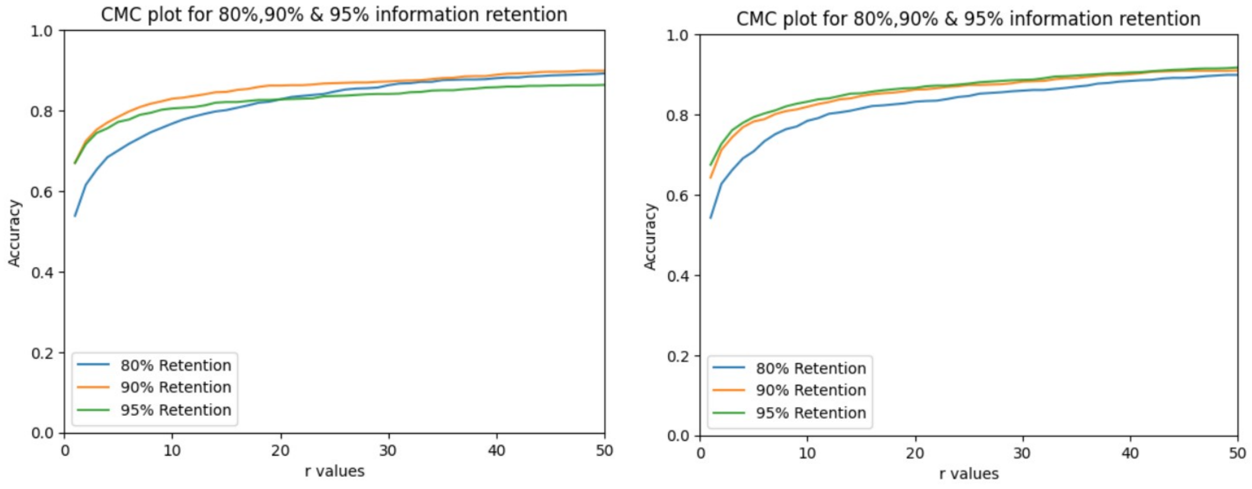
matched correctly for r=1.

2.5 a.IV

In this section, we are displaying the three samples which are incorrectly matched for r=1 meaning the samples which match with the training samples on the first try. In Figure 4a, the first column of images represent the query samples and the second column represent the images with the matched samples. Figure 4b shows example when for r=1 the correct samples were not matched and the output returned different images than the query samples.

2.6 a.V

A comparative study is performed in this section. Three PCA models are trained using 80%, 90% and 95% information retention of the training data. Figure 5a presents a CMC plot with different amounts of information retention and it's performance on the testing data. From the figure it can be seen that for higher and lower value of r information retention of 90% performs the best. 80% of information preservation has better performance for higher values of r over 95% of information preservation. We believe this might have happened because of curse of dimensionality. The number of features while having preserving 80%,90% and 95% information are 15,48 and 113 respectively, so having huge number of features in the last case may have caused to decrease the accuracy. But, we see that when using the Euclidean



(a)

(b)

Figure 5: CMC curve for 80%,90% and 95% when using (a)Mahalanobis distance and (b) Euclidean distance to calculate distance between face coefficients in Face space..

distance to measure the distance in facespace, we are getting in accordance to the amount of information retention (fig 5b). Model with low information retention has low accuracy across the board whereas model with high information retention has high accuracy. So, this also can be because of the way to measure distances.

2.7 b

In this experiment we are removing 50 subjects from the fa_h dataset from the training part but kept those subjects in the testing data. We used PCA with 95% data preservation

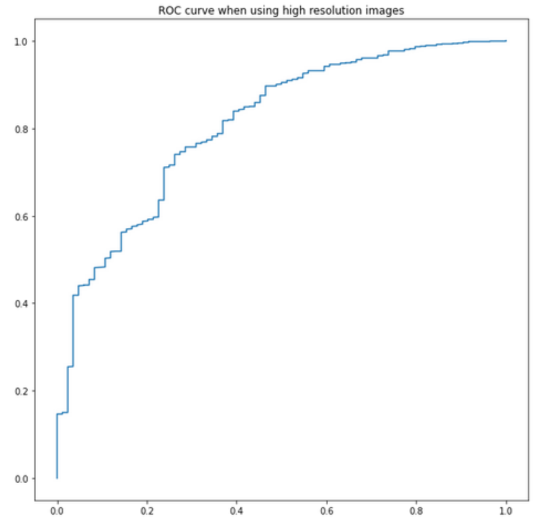


Figure 6: ROC curve for model trained high resolution data.

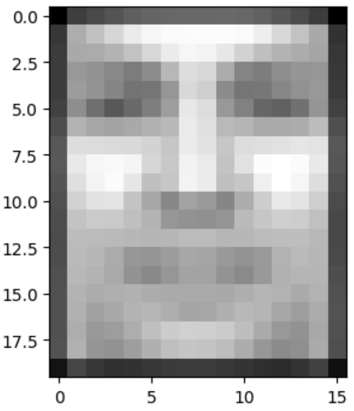
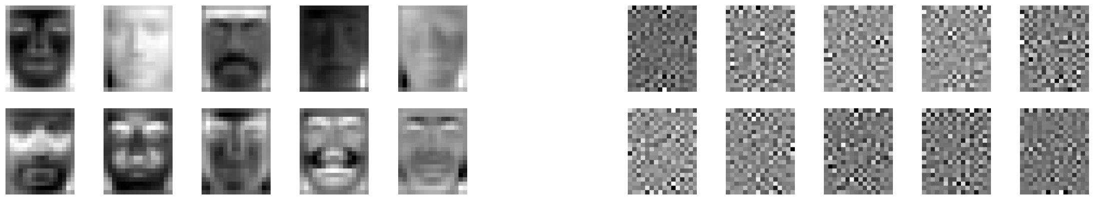


Figure 7: Average face of training image in low resolution.



(a)

(b)

Figure 8: Eigenfaces for (a) largest 10 components and (b) smallest 10 components for training data of Higher resolution.

and tried to detect intruders. We changed the values of threshold between 0 to 1 with 0.01 interval steps and calculated the false positive rates and true positive rates for each threshold values. Using this information we received the ROC curve which can be seen in Figure 6.

2.8 c

All the experiments that have been performed in subsection a is going to be performed here again but with fa_l and fb_l as the training and testing data. The average face (figure 7), largest 10 eigenfaces and smallest 10 eigenfaces (figure 8) corresponding to the eigenvalues for this dataset can be found in Figure 8. The average face is a smoother image of all the peoples faces present in this dataset. This dataset contains images of lower resolutions that is why we are seeing each square pixels and the images are not that smooth. The largest eigenvalues corresponding eigenfaces contains the most information and smallest eigenvalues corresponding eigenfaces contains the least amount of info.

A CMC curve for preserving 80% of the information along on both training data and testing data are shown in Figure 9. From the curve we are seeing that with retaining 80% of the data we are achieving approximately 90% of test accuracy for highest value of r. For r=1 we are obtaining around 50% of accuracy. CMC plot for 80%,90% and 95% of information can be found in Figure 10. For lower values of r, 95% of information preservation performs the best while the 80% information preservation performs the worst. However, for higher

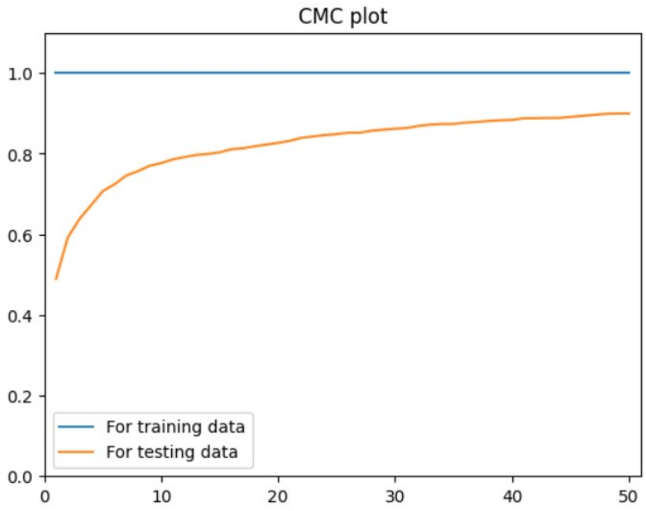


Figure 9: CMC curve for 80% of data.

values of r 90% of information preservation performs better than the one with 95% one. We believe it is due to the curse of dimensionality in higher r values.

2.9 d

The experiment that we performed in subsection b is performed in this stage again but on the low resolution images. All the images of the 50 subjects were removed from the fa_l training dataset but were kept for the fb_l testing. With varying threshold values between 0 to 1 we took 100 intervals and produced the ROC curve given in fig 11.

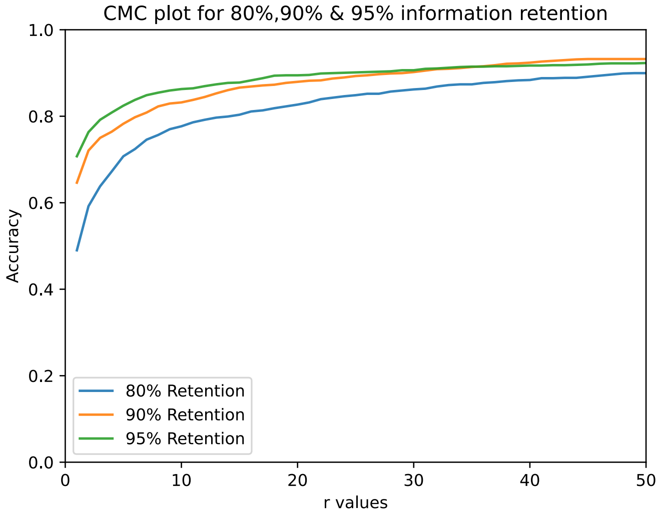


Figure 10: CMC curve for 80%,90% and 95% of information retention for model trained in Lower resolution data.

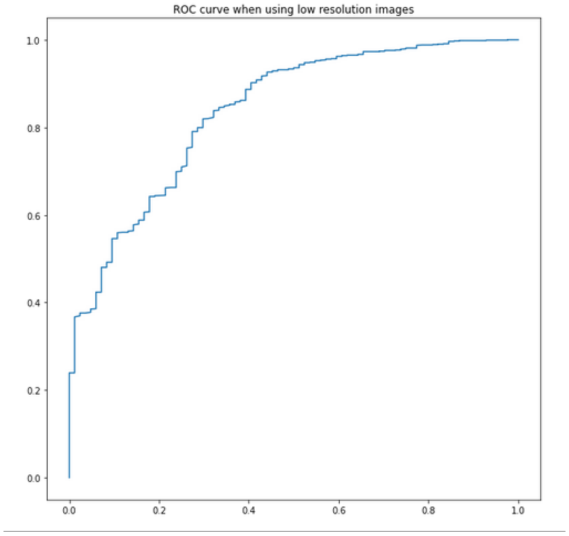


Figure 11: ROC curve for model trained low resolution images.

2.10 e

After feature reduction using PCA with preserving 95% information, it is observed that model trained in low resolution images gave better performance for intruder detection than model trained in high resolution except for a few points where the high resolution did better. We plotted ROC curves for low and high resolution on a same image to better understand the performance for intruder detection. We assume this occured because in low resolution we had fewer features for both training and testing than the high resolution case. PCA tries to minimize information loss but it can't give a definite answer on which features should the model focus more for better results. With less number of features from the low resolution images this problem is faced much less by the classifier thus performing better. A model

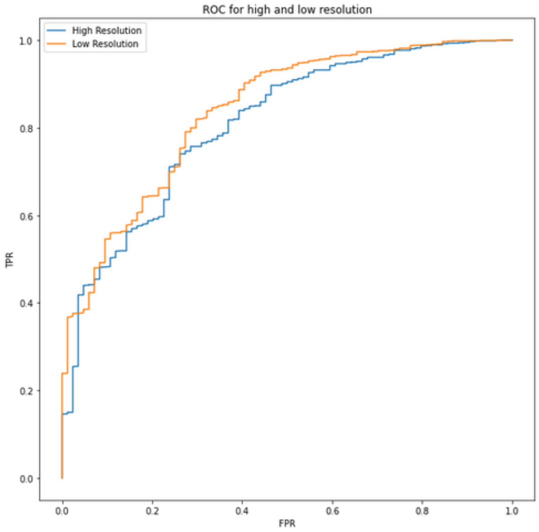


Figure 12: ROC curve for model trained high and low resolution images.

with higher area under the ROC curve (AUC) is understood to have better performance in general. Though we didn't calculate AUC but from Figure12 we can say the model with high resolution have a lower AUC score and overall performed worse than the model trained with low resolution. This shows that in image analysis (and so in other application), better image quality for human eye doesn't correspond to better information for Pattern Recognition model's performance.

3 Part 3: Conclusion

Through the experiments in this assignment we were able to explore the dimensionality reduction capabilities of PCA and with the help of it face recognition task was performed on two different resolution of datasets.