**NANYANG TECHNOLOGICAL UNIVERSITY**

**CSC 419/CPE 433: MACHINE LEARNING**

Semester 2, AY2013/2014

FACE RECOGNITION USING PRINCIPAL COMPONENT ANALYSIS AND LINEAR DISCRIMINANT ANALYSIS

Group ID: 10

Group Members:

Arnav Kumar (U1020584A)

Bindal Dhruv (U1023041F)

Jain Manik (U1022109L)

Keerthana Kumar (U1023092D)

Date of Submission: April 6, 2014

# Abstract

# 1. Introduction

## 1.1. Curse of Dimensionality Problem

The curse of dimensionality refers to the issues associated with multivariate data analysis as the dimensionality increases. As the number of features for machine learning increases and crosses a threshold, the accuracy of the results obtained decreases. The system gets “confused” by the large number of features that need to be taken into account during classification. Additionally, training the system using large number of features is more time and resource consuming. However, using too few features results in generalization, as there are not enough factors to be considered while assigning class labels. There is an evident tradeoff between the dimensions used and final accuracy achieved.

For a given sample size, there is a maximum number of features above which the performance of a given classifier will degrade rather than improve. In most cases, however, the additional information lost by discarding some of the features is compensated by a more accurate mapping in the lower-dimensional space.

There are various ways to overcome this curse of dimensionality problem. Some of the possible methods include:

* incorporating prior knowledge
* providing increasing smoothness of the target function
* reducing the dimensionality

## 1.2. Dimensionality Reduction as a Solution

Dimensionality reduction has proved to be a useful method to overcome the curse of dimensionality problem. Two approaches to perform this are feature extraction and feature selection. Feature extraction involves creating a subset of new features from the entire existing feature set. Feature selection, on the other hand, involves choosing a subset of entire feature set.

Feature extraction techniques are divided into two broad categories - signal representation and signal classification. Feature extraction techniques use the signal representation criterion if the final goal is to represent samples as accurately as possible in lower dimensional space. Feature extraction techniques use the signal classification criterion if the final goal is to enhance the class discriminatory information in lower dimensional space.

## 1.3. What is PCA?

Principal Component Analysis (PCA) is an unsupervised learning method used for dimension reduction, using signal representation. The objective of PCA is to reduce the number of features for machine learning while preserving as much of the variance as possible in high dimensional space. This is done by finding a few orthogonal linear combinations with the largest variance. PCA finds basis vectors for a subspace which maximizes the variance retained in the projected data.

*Advantages*

PCA is a simple method for reducing dimensionality. It is efficient in finding new, uncorrelated, and more informative features. Some of its advantages are listed below.

* It has low sensitivity to noise.
* It has low requirements for memory and capacity.
* It has high efficiency if the processes are taking place in smaller dimensions.

*Disadvantages*

* PCA does not consider label information.
* Moreover, it cannot properly deal with data lying on nonlinear manifolds; it can only find a linear subspace.
* It is also difficult to accurately calculate the covariance matrix.
* Even the simplest invariance can not be captured by PCA unless the training data explicitly provides this information.

## 1.4. What is LDA?

Linear Discriminant Analysis (LDA) is a supervised learning method used for dimension reduction, using signal classification. The primary objective of this method is to preserve as much of the class discriminatory information as possible during this process. In LDA,

*Advantages*

* Multiple dependent variables.
* Reduced error rates
* Easier interpretation of inter-group Differences: each discriminant function measures something unique and different.

*Disadvantages*

* LDA implicitly assumes Gaussian distribution of data.
* It implicitly assumes that the mean is the discriminating factor, not variance.
* It may over-fit the data.

# 2. Problem Definition

## 2.1. Description of the Problem

Facial recognition technology is becoming increasingly important today. With increased security concerns, facial recognition is a useful method to identify potential suspects. With the onset of technology and social media, consumers are beginning to expecting smarter systems that auto-detect faces and take appropriate actions. For example, a camera that automatically captures images based on facial recognition (categorization, and not identification) or automatic picture tagging on popular websites like Facebook that suggest tags for new pictures uploaded, thereby increasing convenience for the end user.

## 2.2. Aim of the Project

This project aims to implement Principal Component Analysis and Linear Discriminant Analysis to perform facial recognition on a databases of images. After basic analysis, for further utility, a user interface is provided to allow users to capture an image using the primary camera attached to their computer and employ the learning algorithms to recognize their faces against a known set of images.

# 3. Methodology and Experimental Settings

## 3.1. Model Selection

For this project, the holdout method was used. The dataset was divided into two sets for training and testing respectively. The former was proportionately larger than the latter. The entire dataset (training and testing sets combined) consisted of 400 images, 10 each of 40 different people. For each person, 8 images were used for training purposes while the remaining 2 were used for testing. The classifier was trained using the training set, and this trained classifier was then applied to the test set in order to predict the resultant label (corresponding face, in this case) to estimate the error rate.

While the holdout method can have issues in case an ‘unfortunate’ split occurs, this effect was expected to be minimal due to the variation in images in either set due to the different faces and expressions. Thus, the holdout method’s disadvantages were mitigated due to the nature of the dataset.

## 3.2. Experimental Setup

The database used was from an archive of AT&T Laboratories Cambridge, called ‘The Database of Faces’. There are a total of 400 images in the database, comprising of 10 images each of 40 different individuals. The conditions vary such that pictures of a single individual may differ in aspects like lighting, facial expressions and facial details like glasses. Each image 92 x 112 pixels in size, with 256 grey levels per pixel.

## 3.3. Classification Approach

The nearest neighbor algorithm is a pattern recognition classification or regression method in which the class label of an object is predicted based on a majority vote. The class most common among the neighboring objects is assigned to the object. In this project, k is taken as 1, i.e., the class of an object depends on its nearest neighbor. Euclidean distance is taken as the distance between two objects, to determine the nearest object.

## 3.4. Comparison Metrics

Once the facial recognition algorithm was trained and subsequently tested for both PCA and LDA, a comparison was performed on the obtained results. The measure used to compare performance was accuracy of the test. Facial recognition for an image was defined to be accurate if the system was able to identify the individual whose image it was. After this was performed for all 80 images in the test set, a comparison was performed and a graph plotting correctly identified images against number of eigenvectors was plotted using Matlab. More information regarding this comparison is discussed in results section of the report.

# 4. Implementation

## 4.1. Data Preprocessing

A class label was assigned to each face image in the database. The Matlab command *imread* was used to read the images, which read a grayscale or colour image from the specified file, and returned an array containing all the image data. In this case, since all the images were in grayscale, they were read as two-dimensional arrays, as opposed to a color image, which may be read as a three-dimensional array, with one two-dimensional array each to represent red, green and blue values for each pixel.

Next, 2D images are converted into vectors. The Matlab *reshape* command was used with parameters such that it returns a N^2 x 1 matrix, where NxN was the size of the image before reshape. More specifically, the N for this analysis was 100, i.e., all the images in the database were first resized to a size of 100x100 pixels.

Once the images were loaded into memory and the following pre-processing steps were performed. First, the images were converted to grayscale with a consistent resolution of 100x100, for the purpose of normalization. Second, images were cropped to only show the face. This was done using the Viola Jones object detection framework. The Viola-Jones approach was developed in 2001 and is used for multiple feature recognition in data. The associated learning algorithm uses AdaBoost (Adaptive Boosting) to select the best features and use them. Boosting refers to using weak classifiers to train a stronger classifier by using a weighted sum approach. AdaBoost is considered adaptive due to its tweaking of subsequent weak learners in iterative computation in favor of those misclassified by previous classifiers. Boosting in general uses several weak learners, such that each weak learner may be only slightly better than a completely random approach, however, the complete model to which the weak learners converge is strong.

## 4.2. Dimensionality Reduction

### 4.2.1. Principal Component Analysis

Column vectors were made out of each of the images in the training and testing image databases. The mean value of the image is then subtracted from each image vector.

The face detector classifier was trained in the following steps:

1. Calculate the mean of the input face images.
2. Obtain the mean-shifted images by subtracting the mean from each of the input images
3. Calculate the eigenvectors and eigenvalues of the resultant shapes
4. For each of the eigenvectors, sort the eigenvalues in decreasing order
5. Retain the principal components (only the eigenvectors with the largest eigenvalues)
6. Using the eigenvectors obtained from the previous step, project the mean-shifted images into the eigenspace.

### 4.2.2. Linear Discriminant Analysis

The LDA object recognition process is an improvement over the PCA method for dimensionality reduction. The goal of LDA method is to maximize the ratio of between-class variance and within-class variance with respect to eigenvector w, i.e., try and maximize between-class variance and minimize within-class variance with respect to the eigenvector.

In a multi-class problem, a generalization of the within-class scatter (SW) is a sum of the within-class scatter of each class. This is shown in the equation below.

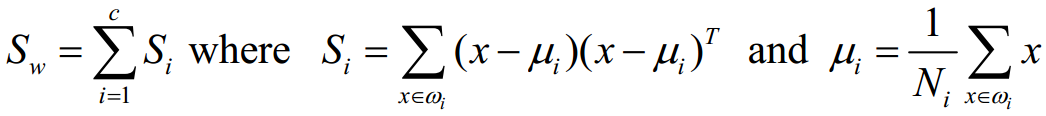


Figure 1 Equation for Calculating Generalization of Within-class Scatter

Additionally, the generalization for the between-class scatter (SB) is as follows.

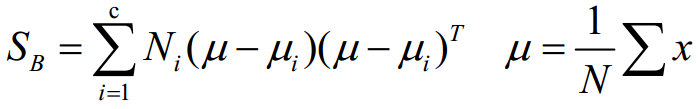


Figure 2 Equation for Generalization of Between-class Scatter

The total scatter matrix is calculcated by ST = SW + SB.

## 4.3. Classification

The similarity between any pair of face images is calculated once the images are projected into the eigenspace. This is done by using the respective feature vectors to find the Euclidean distance ||y1-y2||. The smaller this distance, the more similar the two faces.

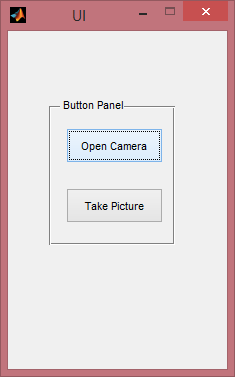
The k-Nearest Neighbor method (with k=1) is then used for classification in the reduced feature space, and the similarity score is calculated between the input image and each of the images in the loaded training set. The face that gives rise to the highest similarity value is taken as the matched face. A similarity value of 1 indicates an exact match.

## 4.4. Performance Evaluation

The final step performed as part of the analysis was to evaluate the results of each of the facial recognition algorithms and compare the results. As part of a more extensive evaluation, the performance evaluation was repeated for a range of eigenfaces.

# 5. Application

An application was developed to demonstrate the image processing framework developed by the team. This application makes use of all the various techniques used for the rest of the project. The figure below shows the user interface of the application.



## 5.1. Preprocessing

A database of images for the various people whose faces need to be detected by the application is created. Since the application crops and prepares the images from the database on the fly during the recognition step, no preprocessing is done on them.

## 5.2. Image Acquisition

Since different computers have different default color space settings for their video cameras, to standardize the input, the stream’s format is set to RGB.

When the user presses the “Open Camera” button, the application starts the video stream and displays it to the user.

The user now needs to press the “Take Picture” button, at which point, an image of the current frame of the video input stream is captured.

This image is converted to grayscale and cached locally. The image below shows the captured image after processing.



## 5.3. Post-Acquisition Processing

Once the image has been acquired and cached, it is the turn of the image processing framework to analyze the image and recognize the faces in it. The following steps detail the process followed.

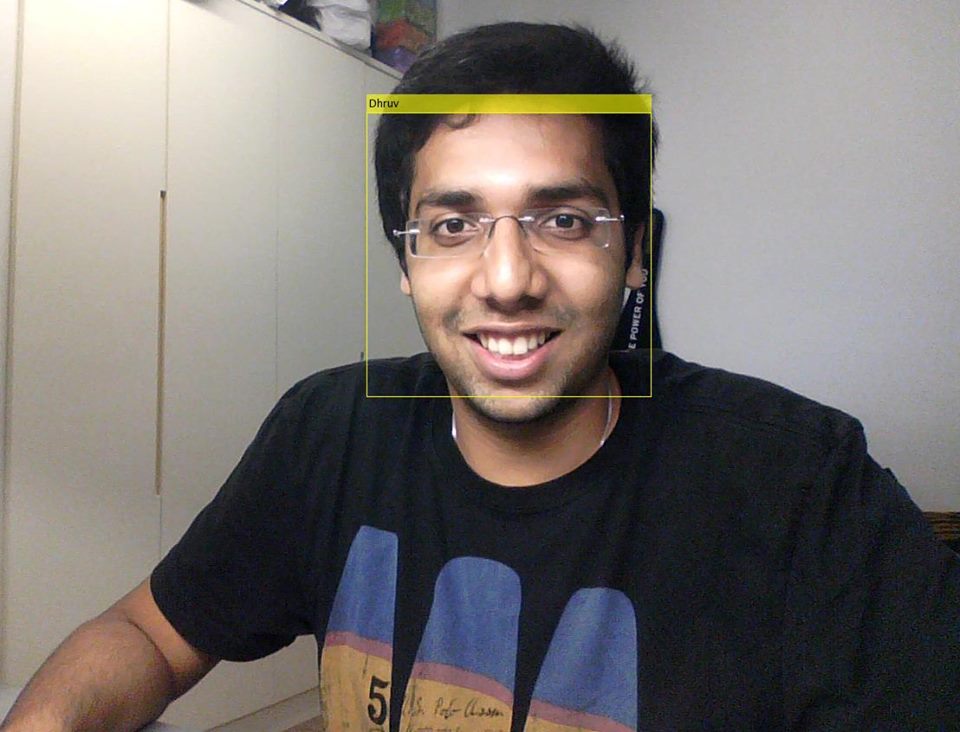
1. All the images from the database are read.
2. These images are preprocessed for the LDA/PCA training step. This is done by converting them to grayscale.
3. After the conversion, all the faces in the photo are extracted.
4. The image recognition algorithm is run on these cropped parts and they are matched against the image database. These cropped faces are inherently eye-localized as well because that is taken care of by the image recognition algorithm.

The below image is the image to which the previous image was matched on analysis.



## 5.4. Results & Achievements

The cropped faces are matched against the database of photos. Once they are recognized, a rectangular box is draw around the face in the image taken by the webcam with a label showing the person’s name. The figure below is the image that the program outputs, with the recognized face in the yellow bounding box.



# 6. Results

The comparison analysis of PCA and LDA was done using the facial image database from AT&T Laboratories Cambridge. The database contains 10 images of the face of 40 different people, resulting in a total database of 400 pictures. These photos have been taken in different locations, with varied lighting and some tolerance for side movement. The main comparisons, shown below reflect the speed and accuracy of the the two types of algorithms.

## 6.1. Analysis of PCA Results

Principal Component Analysis yielded positive results when implemented in MATLAB. For each individual, 9 images were loaded, and their 10th image was used Upon reading the images, following post processing, the following image was the mean image of the entire database:



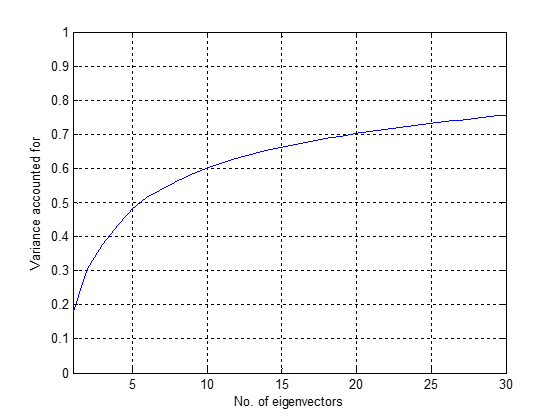
The originally loaded image, along with a shifted image is shown below:

Following the shifting, PCA (as described earlier) was performed and 40 eigenfaces were obtained. The first 5 of them are shown below:



From these 5 eigenfaces, it is shown that each eigenface accentuates a different feature of the face. While some focus more on the eye, others focus more on the mouth or the ears. The following graph shows how much variance in the dataset is accounted for, with increasing number of eigenvectors:



This figure shows that the first eigenface accounts for only 20% of the variance in the dataset, whereas 30 eigenfaces account for over 75% of the variance.

These eigenfaces were then used to generated feature vectors, following which a similarity score was found between the provided input image and the images in the database. The image with the highest similarity score was printed, along with the original image. The provided input face, for facial recognition, and the matched face are shown below:



The execution time (including loading images from the database and training) for PCA was measured for 10 unique runs. The results are shown in the table below:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Speed (sec)** | 20.39 | 19.80 | 20.35 | 20.23 | 19.96 | 19.63 | 20.32 | 20.47 | 19.97 | 20.46 |

## 6.2. Analysis of LDA Results

Linear Discriminant Analysis yielded positive results as well when implemented in MATLAB. For each individual, 9 images were loaded, and their 10th image was used Upon reading the images, following post processing, the following image was the mean image of the entire database:



Upon finding the mean of the entire database, the mean within the same class (for the same person) was found. The mean for 5 different individuals is shown below:



The image scatter between classes for these individuals is shown below:



The image scatter within the same class, for one of the individuals in the database is shown below:



Following all these operations and by performing further calculations, 20 eigenfaces were obtained. The first 5 of those eigenfaces are shown below:



As seen in the case of PCA, even the LDA eigenfaces accentuate certain parts of the face and hence helped in the analysis. Finally, upon projecting the input test image and all the images in the training set with the eigenfaces, the distance between the input image’s projection and the images in the database was found, and the image with the minimum distance was printed as the matched face. The input test image, followed by the recognized image is shown below:

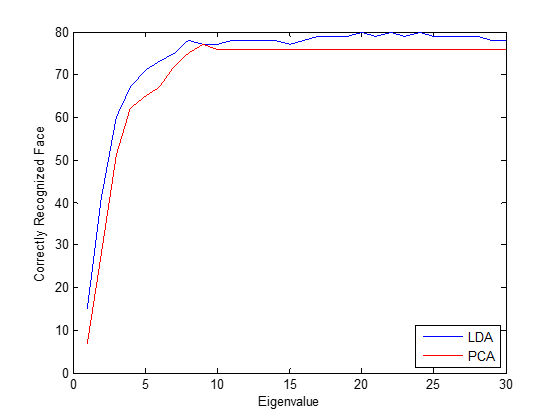


The execution time (including loading images from the database and training) for LDA was measured for 10 unique runs. The results are shown in the table below:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Speed (sec)** | 3.40 | 2.51 | 2.57 | 2.83 | 2.57 | 2.53 | 2.56 | 2.53 | 2.64 | 2.55 |

## 6.3. Comparison of PCA and LDA

To compare PCA and LDA, 8 images of each individual were loaded into the training set and two of them were kept for the testing set. Thus there were 320 images in the training set and 80 images in the testing set. By starting with an eigenvalue number of 1 and increasing it till 30, the accuracy of both PCA and LDA was tracked. The resulting graph is shown below:



From the chart, it is clearly observable that LDA was consistently better than LDA for all eigenvalues. Furthermore, while the PCA analysis peaked at 76 successful recognitions (thus it always failed with 4 of them), LDA had 100% accuracy (or thereabouts) when the eigenvalue was 20 and above. Hence it was found that LDA resulted in a higher accuracy rate than PCA for dimensionality reduction.

On comparing the speed of execution of PCA and LDA, it was also observed that LDA was 20 seconds faster than PCA in execution. However it is important to note that this result is largely implementation specific and can vary.

From both the above observations, including accuracy and speed of execution, it was seen that LDA performed better than PCA.

# 7. Conclusion

## 7.1. Summary of Project Achievements

The project was successful in implementing Principal Component Analysis and Linear Discriminant Analysis on a database of known images derived from the AT&T archive. The authors proved to go one step further by implementing the facial recognition algorithms to recognize faces of team members using a set of images compared against an image taken from the webcam, thereby exhibiting real life utility of the developed system.

## 7.2. Limitations and Recommendations

Although the team was able to meet the defined objectives by using PCA and LDA for face recognition and comparing the final classification accuracy obtained by both, there were some limitations. Further research and experiments could be conducted in the future to explore other areas to obtain more comprehensive and solid results for classification accuracy in facial recognition. For future development, other subspace-based face recognition algorithms such as Independent Component Analysis (ICA) and Non-negative Matrix Factorization (NMF) could be implemented. The former aims to maximize statistical independence of the dimensions while selecting subspace projections, while the latter aims to generate non-negative basis vectors. These methods may give rise to a better classification accuracy rate.

# 8. References

https://engineering.purdue.edu/kak/computervision/ECE661\_Fall2012/solution/hw9\_s1.pdf

http://blog.cordiner.net/2010/12/02/eigenfaces-face-recognition-matlab/

https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

“An Overview of Principal Component Analysis” by Sasan Karamizadeh, Shahidan M. Abdullah, Azizah A. Manaf, Mazdak Zamani, Alireza Hooman.

http://people.eku.edu/falkenbergs/psy862/notes%20on%20discriminant%20analysis.html