Abstract

The aim of the project is to study the dimensionality reduction techniques focusing on their applications in face recognition and representation. The primary task in face recognition applications is to represent the facial data in lower dimensionality space and finally work in the lower dimensional space to meet the requirements of face recognition. PCA also sometimes called as linear PCA is one such method which projects the face images onto the eigen space, also called the face space, which is a lower dimensional representation of the face images. I have used the Mahalanobis distance to meet the requirements of face recognition. LDA is another such technique which projects the facial images onto what is called a Fisher space which maximizes the ratio of the between class cluster to within class cluster. This performs better in classification than PCA. For classifications in higher dimensions the kernel methods are adopted namely the kernel PCA(KPCA) and the kernel Fisher Discriminant(KFD). I have taken the polynomial kernel to implement KPCA and KFD in face recognition problem. UMIST face databases has been used for implementation.

Index Terms: Principal Component analysis (PCA), Kernel Principal Component Analysis(kPCA), Linear Discriminant Analysis(LDA), Multiple Discriminant Analysis(MDA), Kernel Fisher Discriminant, Locally Linear Embedding(LLE), UMIST database, Eigen Faces, Fisher Faces, Nearest Neighborhood(NN) classifier

1. Introduction

Advances in data collection and storage capabilities have led to a tremendous information overload. As a direct consequence this led to the collection of data samples represented as real-valued vectors having too many dimensions.

One might desire a higher dimension of this real-valued vector as this might improve the classification power of the classifier. However, this increases the computational complexity. This is often known as the "the curse of Dimensionality. Hence the need arises where an optimal method needs to
be found out such that the information gathered from the samples from the higher dimensions actually is preserved and can be faithfully interpreted in low dimensional space.

Any Dimensionality Reduction technique can be broadly classified as Linear Methods and Non Linear Methods. The linear methods like the PCA ,LDA are direct linear manipulations of the input data so as to represent the higher dimensional data in lower dimensions. The non-linear techniques however do not have a direct linear manipulations of the data to aid in lower dimensional spaces. The techniques using the kernel matrices like the KPCA, KFD fall into this category. Other methods like the LLE and ISOMAP also are methods of non-linear dimensionality reduction techniques.

Face Recognition is a high dimensional pattern recognition problem. Even low-resolution face images generate huge dimensional feature spaces (20,000 dimensions in a small 100x200 pixels). In addition to the problems of large computational complexity and memory storage, this high dimensionality makes very difficult to obtain statistical models of the input space using well-defined parametric models. In this project I have applied the PCA, LDA ,the kernel based equivalent approaches of PCA and LDA ,namely the KPCA, KFD and applied them to address the face recognition problem.

I have used the UMIST database for face images and applied some of the image processing techniques like histogram equalization, noise removal due to variations in lighting conditions and orientation of images, in the UMIST database and used the processed images for implementing the dimensionality reductions techniques to solve face recognition problems.

2. Study Phase of the Project

As the first phase of the project I have studied the dimensionality reduction techniques namely the PCA ,LDA, KPCA, KLDA .This study phase also involved the study of classifiers namely the NN classifier ,a classifier with k=1 in k-nearest neighbor classifier. A part of the study was exclusively towards the face databases .The goal of the latter study was to find the reliable face databases to actually use them in this project. Certain techniques like the histogram equalization techniques were also studied to get a better representation of the face images so as to reduce the error rate of classification.

The findings of the above study are summarized in the following sections

2.1 Study of Dimensionality Reduction Techniques

2.1.1 PCA : Principal Component Analysis

PCA assumes that the information is carried in the variance of the features: the higher the variance in one dimension (feature), the higher the information carried by that feature .Thus the transformation is based on preserving the most variance in the data using the least number of dimensions. The data is projected onto a lower dimensional space where
the new features best represent the old features in the least squares sense.

PCA involves the computation of the eigen vectors of the covariance matrix of the given data set as shown by the following equation

$$XX^T e = \lambda e \quad (1)$$

Here $e$ is the eigen vector and $\lambda$ is the eigen value. We can now multiply both sides by $X^T$ to give a new equation

$$K\alpha = \lambda \alpha \quad (2)$$

Where $K = X^TX$ and $\alpha = X^Te$. K here is called as an inner product matrix such that $K_{i,j} = \langle x_i, x_j \rangle$. The principal components $y = e^T x = \sum_i^N \alpha_i \langle x_i, x \rangle$ for $x$ being the test vector. In this project $x$ is a Face Image of one person. We then arrange the eigen vectors in a decreasing order of the magnitude of the corresponding eigen values and reject the eigen vectors which correspond to the eigen values having a low magnitude. This is the main trick in dimensionality reduction. In this method since eigen vectors correspond to images these vectors are also known by the name eigen Faces. Hence this method is called as eigen Face Based Approach.

2.1.2 LDA (Linear Discriminant Analysis) or the FDA (Fisher Discriminant Analysis)

PCA finds the minimum number of components that best represents the data. However, this best representation is in the least square sense. It does not guarantee any usefulness for discrimination / classification. We need to reduce the dimensionality, under some constraint of maximizing the class discrimination. FDA is in short the following projection:

$$Y = W^T X$$

Each $Y_i$ being the projection of $X_i$ in the direction of $W_i$.

Maximizing the discrimination between classes can be achieved by increasing the intercluster distances and reducing the intracluster distances. These distances are obtained using between and within-class scatter matrices, as shown below.

$$S_w = \sum_{i=1}^{c} S_i \quad \text{Total Within Class scatter matrix}$$

$$k(x, x') = \langle \Phi(x), \Phi(x') \rangle \quad \text{Within Class Scatter Matrix for class one class } \omega_i$$

$$S_B = \sum_{i=1}^{c} n_i(m_i - m)(m_i - m)' \quad \text{Between Class Matrix}$$

$n_i$ being the number of elements in each class, $c$ is the number of classes, $m_i$ is the mean of the class $\omega_i$.
The required set of vectors namely $W_i$ are the ones that maximize the function $J(w) = (W^T S_y W) / (W^T S_y W)$. It can be found that the vectors $W_i$ are the eigenvectors of the matrix $S_y^{-1} S_x$. Hence the optimal set of these eigenvectors arranged in a decreasing order of magnitude give the optimal directions for data projection.

### 2.1.3 The kernel non-linear KPCA and the KFD Algorithms.

Kernel based Algorithms do a mapping of input data to a higher dimensional data. Basically this mapping is a very non-linear one. This projection into higher space is done using simple dot products of the input vectors rather than explicitly computing the mapping. Thus this reduces the complexity of the computation by a large amount. The basic concept of KPCA is to first map the input data $x$ into a feature space $F$ via a non-linear mapping $\Phi$ and then perform linear PCA in $F$. A simple yet safe assumption is that the data set of face images is highly non-linear and doing linear PCA after mapping to higher dimensions helps in classification.

Let $X$ denote the input sample and $\Phi(x)$ be the corresponding data vector in higher dimensional space. It is to be ensured that $\Phi(x)$ is properly centered. In other words

$$\sum_i^n \Phi(x_i) = 0$$

A kernel matrix $K$ such that $k = \Phi^T \Phi$ with $K_{ij} = \langle \Phi(x_i), \Phi(x_j) \rangle$. Then the principal directions satisfying (2) are given by $\Sigma \Phi \alpha$. It is a must to normalize $\alpha$. For a given test vector $x$ the principal component $y$ corresponding to eigen vector $e$ is given by

$$y = \sum_i^N \alpha_i \langle \Phi(x_i), \Phi(x) \rangle.$$

The dot product can be computed by choosing a kernel $k(x,y)$ such that $k(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle = K_{ij}$.

This avoids the computational complexity and is sometimes called the Kernel Trick. The kernel used in this matrix is a polynomial kernel given by $k(x,y) = (1 + \langle x, y \rangle)^P$, $P$ being the degree of the polynomial.

For a kernel version of LDA namely KF, an exactly similar approach is employed. As from the theory of LDA, it can be easily predicted that KLDA provides an excellent class recognition of all the above algorithms described above.

### 2.1.4 Classifier selection

In order to classify the faces which are being mapped onto the higher dimensional space using the kernel trick (as in KPCA), I have used the simple NN classifier to aid in classification. A simple NN classifier also called the nearest neighborhood classifier assigns a label to the data element basing on the class of its nearest neighbor. In other cases like the KFD NN classifiers may not be used as KFD itself is a powerful tool to help in classification.
2.2 Face Databases

I have studied the existing face databases, and the way the face images are being stored in the databases. A face is usually represented as a 2-dimensional image and the size of the face can vary. Some databases provide the cropped versions face images. UMIST is one such database providing the cropped versions and it is being used in the project. The image size is 92*112. However, certain modifications need to be done on the picture like the histogram equalization so that the picture has a uniform contrast. This is required because the classifier might not perform well if the image is too dark or if the lighting conditions are too much on one side and not on the other. Thus, lighting conditions in the images is another aspect which needs to be taken into consideration while classifying. Other databases which I encountered were the YALE database and the MIT face database, FERET database (which has its images histogram equalized).

3. Implementation

In this phase, I will be implementing the above studied techniques on the face images in MATLAB. The UMIST database is used. Face images of five different persons showing 15 different orientations is used as a training set.

3.1 The Eigenface Approach

I have implemented the first technique namely the Face recognition using EigenFaces in Matlab. The implementing algorithm can be summarized as follows:

Face Representation:
- A mean face is first found which is the average of all the face pictures. This is easy to find in MATLAB.
- I have subtracted the mean from each of the individual images to give a new set of face vectors.
- The covariance matrix $XX^T$ is not calculated but alternatively the eigen vectors $V_i$ of $XX^T$ is found. The M largest eigen vectors, $U_i$ of the covariance matrix is found by the equation $U_i = XX^T V_i$. This considerably reduced the computations. Face recognition:
- The input vector is used to compute the Mahalanobis distance. This is also a measure of the error. The one sample in the training set which has the least error is found out to be the one as the input vector. If the error falls beyond a threshold then none of the vectors in the input training set matches the input vector.

Output of the eigen Face approach:
A successful recognition is represented as a similar image being regenerated. A failure is determined as either a false image generation or a another class image regeneration.
3.2 The Fisher Based Approach

This method is similar to the previous one and as explained in previous sections it computes the eigen vectors of the product of two matrices. One being the inverse of the within scatter matrix and the other being the between class matrix. Since each image was 92*112 size, the number of dimensions for a test input was 10304. Hence the within scatter matrix was of the order 10304*10304 which was computationally highly complex. Hence an alternate strategy was used. I made use of the fact that if the image can somehow be converted into the frequency space by applying a suitable transform then the coefficients in the lowest frequency can be used to represent the image. Hence I have applied DST to the images as they are fed. The DST function can be had from the image processing toolbox in matlab or can be implemented very easily. Thus the input vector is reduced to one having about 30*30 DST coefficients. After this transformation the in-class and within class matrices are computed and a reduced dimensional space was found. The recognition procedure was similar to the one used for eigen Faces.

3.3 The kernel PCA and kernel LDA or KFD algorithm implementations

KPCA was very similar to the PCA algorithm. The kernel trick was applied by taking the kernel matrix

\[ k(x, y) = (1 + \langle x, y \rangle)^p \]  

After this step the process is exactly similar to PCA. The KFD implementation as expected will be exactly similar to LDA after the application of the kernel trick. One main criterion that needs to be taken care here is that data in higher dimension needs to be centered. If not face recognition falls drastically.
4. Results

The results obtained can be summarized in Table. The criterion for comparison of the algorithms was simple. The error rate was calculated as the percentage of the ratio of failed recognitions to that of the number of trials. All these methods are central to the number of eigen vectors taken. Since as an input we have 85 images the number of eigen values vary from 1 to 85. The more the eigen vectors the lesser the error rate.

It is observed that KFD performs better than all other algorithms. When a minimal set of eigen vectors are given the variations in the error rates of the four algorithms are predominant and as the number of eigen vectors are increased the performance of all the algorithms is similar. Thus if the number of eigen vectors are between the range 20-40 we see that KFD performs best . the least performed is the PCA. Since we see that a minimal set of eigen vectors give a very high dimensionality reduction we conclude that implementation of KFD is the best possible method. However the computation time of KFD was the highest .

Another criterion for the study of the kernel based methods is the degree of the polynomial function. Since a polynomial kernel is used in this project ,various values are given to d and the performance is measured. Fig 3 and fig 4 show this variation .It can be easily concluded from the graphs that as we increase the degree of the polynomial kernel the error rate increases. However as the number of eigen vectors increase the performance becomes better even with increasing values of d .but at the expense of more computation time.

Fig2. Plot showing the error rate as a function of the Number of eigen vectors used for image regeneration.
Fig3. Plot showing the variation of error rates as a function of eigen values for various values of $d$.

The variations in KFD for degrees greater than 2 is not performed due to the computational complexity of KFD algorithm.

The complexity of the algorithms are studied taking the computation time into account. Fig4 gives a comparative study of the complexity of the algorithms. The algorithms were applied to a same test image from an UMIST dataset and the outcome was a success.

Fig.4 Computation time of various algorithms
5. Conclusion:

Dimensionality reduction techniques play a major role in pattern classification problems especially when the dimensions of the data to be clustered is very high. This project clearly highlights the above fact. Face databases are indeed very high dimensional spaces and clustering without dimensionality reduction is almost an impossible job. The project provides a comparative study of various dimensionality reduction techniques. It can be concluded basing on the results shown in the previous section that kernel based algorithms perform better than their linear counterparts. Also if the projection of data is onto a minimal set of dimensions, KFD algorithms perform better followed by KPCA, LDA and PCA respectively. However LDA and KFD algorithms are more complex as they involve computation of the within and between class scattering matrices. Also the kernel matrix is also studied and it is seen that as the degree of polynomial is increased the kernel methods show a slightly degraded performance.

6. Future Work

Projection of data onto a lower dimensional space is an active area of research. Kernel based methods is one such area which provides optimum classification of data. All the existing non-kernel based algorithms can find their kernel equivalent approaches in near future. KFD and KPCA are two such examples. Also classification based on a combination of two or more kernel based techniques can be applied on high dimensional data, if a very high level of accuracy is a must.

If the dimensionality reduction techniques are targeted on solving face recognition problems the need for a standardized face databases is mandatory. FERET is a very recent face database which has images showing little variations in lighting conditions of the facial image. Thus such histogram equalized databases provide a better understanding and study of the dimensionality reduction techniques with a greater level of accuracy.
References:


Web Links:

Face Databases:
www.face-rec.org/databases
http://images.ee.umist.ac.uk/danny/database.html