We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



7

Dimensionality Reduction Techniques for Face Recognition

Shylaja S S, K N Balasubramanya Murthy and S Natarajan
Department of Information Science and Engineering, P E S Institute of Technology India

1. Introduction

High level of image content analysis is required for several applications. This is taking more significance as the number of digital images stored is growing exponentially. On the one hand the technology should help store these images, on the other, enable us to develop newer algorithmic models aimed at efficient and quick retrieval of images. The entire captured data may not be applicable for an application and hence deriving a subset of data to achieve objective function is desirable.

Face detection and recognition are preliminary steps to a wide range of applications such as personal identity verification, video-surveillance, facial expression extraction, gender classification, advanced human and computer interaction. A face recognition system would allow user to be identified by simply walking past a surveillance camera. Research has been devoted to facial recognition for years and has brought forward algorithms in an attempt to be as accurate as humans are.

A face recognition system is expected to identify faces present in images and videos automatically. It can operate in either or both of two modes:

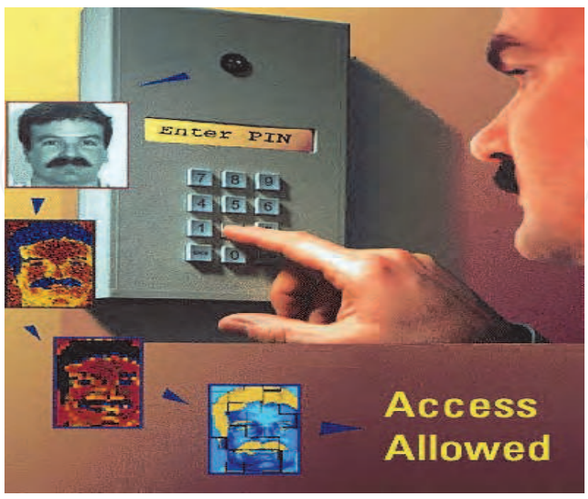


Fig. 1. Face Verification System

- Face verification or authentication,(fig above)

- Face identification or recognition.

Face verification involves a one-to-one match that compares a query face image against a template face image whose identity is being claimed. Face identification involves one-to-many matches that compare a query face image against all the template images in the database to determine the identity of the query face. Another face recognition scenario involves a watch-list check, where a query face is matched to a list of suspects (one-to-few matches). As per Hietmeyer, face recognition is one of the most effective biometric techniques for travel documents and scored higher on several evaluation parameters.

Computational models of face recognition must address several difficult problems. This difficulty arises from the fact that faces must be represented in a way that best utilizes the available face information to distinguish a particular face from all other faces. The problem of dimensionality reduction arises in face recognition because an m X n face image is reconstructed to form a column vector of mn components, for computational purposes. As the number of images in the data set increases, the complexity of representing data sets increases. Analysis with a large number of variables generally consumes a large amount of memory and computation power.

2. Dimensionality reduction

Efforts are on for efficient storage and retrieval of images. Considerable progress has happened in face recognition with newer models especially with the development of powerful models of face appearance. These models represent faces as points in high-dimensional image spaces and employ dimensionality reduction to find a more meaningful representation, therefore, addressing the issue of the "curse of dimensionality". Dimension reduction is a process of reducing the number of variables under observation. The need for dimension reduction arises when there is a large number of univariate data points or when the data points themselves are observations of a high dimensional variable. The key observation is that although face images can be regarded as points in a high-dimensional space, they often lie on a manifold (i.e., subspace) of much lower dimensionality, embedded in the high-dimensional image space. The main issue is how to properly define and determine a low-dimensional subspace of face appearance in a high-dimensional image space.

Dimensionality reduction techniques using linear transformations have been very popular in determining the intrinsic dimensionality of the manifold as well as extracting its principal directions. Dimensionality reduction is an effective approach to downsizing data. In statistics, dimension reduction is the process of reducing the number of random variables under consideration, $R^N \rightarrow R^M$ (M<N) and can be divided into feature selection and feature extraction.

Feature selection is choosing a subset of all the features

$$[x_1\ x_2 \ldots x_n] \xrightarrow{\text{Feature selection}} [x_{i1}\ x_{i2} \ldots x_{im}]$$

Feature extraction is creating new features from existing ones

$$[x_1\ x_2 \ldots x_n] \xrightarrow{\text{Feature extraction}} [y_1\ y_2 \ldots y_m]$$

In either case, the goal is to find a low-dimensional representation of the data while still describing the data with sufficient accuracy.

For reasons of computational and conceptual simplicity, the representation is often sought as a linear transformation of the original data. In other words, each component of the

representation is a linear combination of the original variables. Well-known linear transformation methods include principal component analysis, factor analysis, and projection pursuit. Independent component analysis (ICA) is a recently developed method in which the goal is to find a linear representation of nongaussian data so that the components are statistically independent, or as independent as possible. Such a representation seems to capture the essential structure of the data in many applications, including feature extraction and signal separation.

Several techniques exist to tackle the curse of dimensionality out of which some are linear methods and others are nonlinear. PCA, LDA, LPP are some popular linear methods and nonlinear methods include ISOMAP & Eigenmaps. PCA and LDA are the two most widely used subspace learning techniques for face recognition. These methods project the training sample faces to a low dimensional representation space where the recognition is carried out. The main supposition behind this procedure is that the face space (given by the feature vectors) has a lower dimension than the image space (given by the number of pixels in the image), and that the recognition of the faces can be performed in this reduced space**.** PCA has the advantage of capturing holistic features but ignore the localized features. Fisher faces from LDA technique extracts discriminating features between classes and is found to perform better for large data sets. Its shortcoming is that of Small Sample Space (SSS) problem. LPPs are linear projective maps that arise by solving variational problem that optimally preserves the neighborhood structure of the data set.

In many cases, face images may be visualized as points drawn on a low-dimensional manifold hidden in a high-dimensional ambient space. Specially, we can consider that a sheet of rubber is crumpled into a (high-dimensional) ball. The objective of a dimensionality-reducing mapping is to unfold the sheet and to make its low-dimensional structure explicit. If the sheet is not torn in the process, the mapping is topology-preserving. Moreover, if the rubber is not stretched or compressed, the mapping preserves the metric structure of the original space.

PCA is guaranteed to discover the dimensionality of the manifold and produces a compact representation. Turk and Pentland use Principal Component Analysis to describe face images in terms of a set of basis functions, or “eigenfaces”. LDA is a supervised learning algorithm. LDA searches for the project axes on which the data points of different classes are far from each other while requiring data points of the same class to be close to each other. Unlike PCA which encodes information in an orthogonal linear space, LDA encodes discriminating information in a linear separable space using bases are not necessarily orthogonal. It is generally believed that algorithms based on LDA are superior to those based on PCA. However, some recent work shows that, when the training dataset is small, PCA can outperform LDA, and also that PCA is less sensitive to different training datasets.

Recently, a number of research efforts have shown that the face images possibly reside on a nonlinear submanifold. However, both PCA and LDA effectively see only the Euclidean structure. They fail to discover the underlying structure, if the face images lie on a nonlinear submanifold hidden in the image space. Some nonlinear techniques have been proposed to discover the nonlinear structure of the manifold, *e.g.* Isomap, LLE and Laplacian Eigenmap. These nonlinear methods do yield impressive results on some benchmark artificial data sets. However, they yield maps that are defined *only* on the training data points and how to evaluate the maps on novel test data points remains unclear.

3. Singular Value Decomposition (SVD)

Singular value decomposition (SVD) is an important factorization of a rectangular real or complex matrix, with many applications in signal processing and statistics. As applied to face recognition this technique is used to extract the holistic global features of the training set SVD is the best, in the mean-square error sense, linear dimension reduction technique. Being based on the covariance matrix of the variables, it is a second-order method. SVD seeks to reduce the dimension of the data by finding a few orthogonal linear combinations of the original variables with the largest variance.

The basic idea behind SVD is taking a high dimensional, highly variable set of data points and reducing it to a lower dimensional space that exposes the substructure of the original data more clearly and orders it from most variation to the least. What makes SVD practical for pattern recognition applications is that one can simply ignore variation below a particular threshold to massively reduce the data but be assured that the main relationships of interest have been preserved.

Singular value decomposition (SVD) can be looked at from three mutually compatible points of view. On the one hand, we can see it as a method for transforming correlated variables into a set of uncorrelated ones that better expose the various relationships among the original data items. At the same time, SVD is a method for identifying and ordering the dimensions along which data points exhibit the most variation. This ties into the third way of viewing SVD, which is that once we have identified where the most variation is, it's possible to find the best approximation of the original data points using fewer dimensions. Hence, SVD can be seen as a method for data reduction.

As said earlier Singular Value Decomposition is a way of factoring matrices into a series of linear approximations that expose the underlying structure of the matrix. If A is the input matrix, calculating the SVD consists of finding the eigenvalues and eigenvectors of AA^T and A^TA. This yields three matrices U,V & S where the eigenvectors of A^TA make up the columns of V , the eigenvectors of AA^T make up the columns of U. and the singular values in **S** are square roots of eigenvalues from AA^T or A^TA. The singular values are the diagonal entries of the S matrix and are arranged in descending order. The singular values are always real numbers. If the matrix A is a real matrix, then U and V are also real.

In the factorization, the first principal component is s1, with the largest variance is the linear combination with T T . We have $S_1 = XW_1$, where the p-dimensional coefficient vector solves $W_1 = (W_{11},....,W_{1P})$ where

$$W_1 = \arg\max_{\|w=1\|} Var\{x, w\} \tag{1}$$

The second PC is the linear combination with the second largest variance and orthogonal to the first PC, and so on. There are as many PCs as the number of the original variables. PCs explain most of the variance, so that the rest can be disregarded with minimal loss of information. Since the variance depends on the scale of the variables, it is customary to first standardize each variable to have mean zero and standard deviation one. After the standardization, the original variables with possibly different units of measurement are all in comparable units.

The mathematical model formulated is given below:

Let A is m′ X n′ real matrix and $N=A^TA$



Fig. 2. Range and Null space of matrix

R denotes the range space and N denotes the null space of a matrix. Rank of A, A^T, A^TA, AA^T is equal and is denoted by ρ orthonormal basis v_i $1 \le i \le \rho$ are sought for $R_A{}^T$ where ρ is the rank of $R_A{}^T$& u_i $1 \le i \le \rho$ for R_A such that,

$$AV_j = S_j U \tag{2}$$

$$A^T U_j = S_j V_j \ , \qquad 1 \le j \le \rho \tag{3}$$

Advantages of having such a basis are that geometry becomes easy and gives a decomposition of A into ρ one-ranked matrices. Combining the equations (2) & (3)

$$A = \sum S_j V_j \, U_j{}^T \ , \qquad 1 \le j \le \rho \tag{4}$$

If V_j is known then , $U_j = (1/S_j) A V_j$ $|S_j| = \| AV_j \|$ therefore ,$s_j \neq 0$, choosing $s_j > 0$,

$$AV_j = S_j U_j \tag{5}$$

$$A^T A V_j = S_j A^T U_j \tag{6}$$

$$A^T A V_j = S_j{}^2 V_j \tag{7}$$

Let $S_j{}^2 = \mu_i$, $NV_j = \mu_i V_i$ is required U_i's as orthonormal eigenvectors of $N=A^TA$ are found and $S_j = \sqrt{\mu_i}$ Where $\mu_i > 0$ are eigen values corresponding to V_j. The resulting Ui span the Eigen subspace. When SVD is applied to the sample set below in figure 3, the corresponding eigen faces obtained are shown in figure 4. The figure is highlighting the holistic features from the given sample set.



Fig. 3. Training set example faces



Fig. 4. Eigen faces from SVD

Basis selection from SVD

If A is the face Space, then x vectors are drawn from $[X_1 \ldots X_x] = \Pi_{<1..x>}(A^{-1}UD)$ Where U & D are the unitary and diagonal matrices of SVD of A.

5. Linear Discriminant Analysis

Fisher Linear Discriminant also referred as Linear Discriminant Analysis is a classical pattern recognition method, which was introduced by Fisher (1934). It is a very effective feature extraction method but facing issues for Small Sample Space problem.

The Dimensionality Reduction technique SVD searches for directions in the data that have largest variance and subsequently project the data onto it. In this way, one can obtain a lower dimensional representation of the data, that removes some of the ”noisy” directions. There are many difficult issues with how many directions one needs to choose. It is an unsupervised technique and as such does not include label information of the data. For instance, if we imagine 2 clusters in 2 dimensions, one clusters has $y = 1$ and the other $y = {}_{i}1$. The clusters are positioned in parallel and very closely together, such that the variance in the total data-set, ignoring the labels, is in the direction of the clusters. For classification, this would be a terrible projection, because all labels get evenly mixed and will destroy the useful information.

A much more useful projection is orthogonal to the clusters, i.e. in the direction of least overall variance, which would perfectly separate the data-cases (obviously, we would still need to perform classification in this 1-D space).

The conventional solution to misclassification for small sample size problem and large data set with similar faces is the use of PCA into LDA i.e. fisher faces. PCA is used for dimensionality reduction and then LDA is performed on the lower dimensional space. Discriminant analysis often produces models whose accuracy approaches complex modern methods. The target variable may have two or more categories. The following figure 5 shows a plot of the two categories with the two predictors on orthogonal axes:

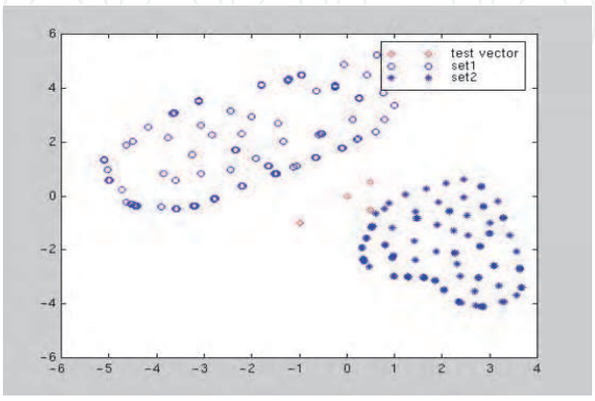


Fig. 5. A plot of the two categories with the two predictors on orthogonal axes

A transformation function is found that maximizes the ratio of between-class variance to within-class variance as illustrated by this figure 6 produced by Ludwig Schwardt and Johan du Preez:

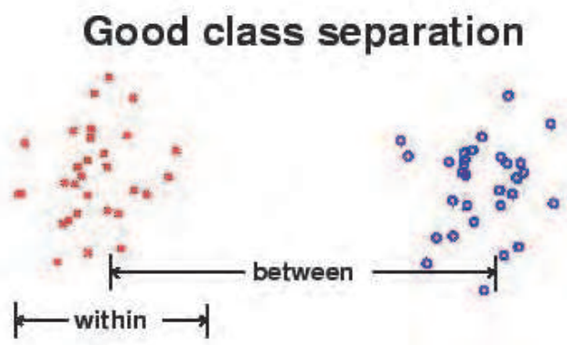


Fig. 6. Output of applying transformation function

The transformation seeks to rotate the axes so that when the categories are projected on the new axes, the differences between the groups are maximized. So the question is, how do we utilize the label information in finding informative projections?
To that purpose Fisher-LDA considers maximizing the following objective:

$$J(w)=\frac{w^T S_B w}{w^T S_w w} \tag{8}$$

The second use of the term LDA refers to a discriminative feature transform that is optimal for certain cases [10]. This is what we denote by LDA throughout this paper. In the basic formulation, LDA finds eigenvectors of matrix

$$T=S_w^{-1}\ S_b \tag{9}$$

Here S_b is the between-class covariance matrix, that is, the covariance matrix of class means. Sw denotes the within-class covariance matrix, that is equal to the weighted sum of covariance matrices computed for each class separately. S_w^{-1} captures the compactness of each class, and S_b represents the separation of the class means. Thus T captures both. The eigenvectors corresponding to largest k eigenvalues of T form the rows of the transform matrix w, and new discriminative features d_k are derived from the original ones d simply by

$$d_k=W_d \tag{10}$$

The straightforward algebraic way of deriving the LDA transform matrix is both a strength and a weakness of the method. Since LDA makes use of only second-order statistical information, covariances, it is optimal for data where each class has a unimodal Gaussian density with well separated means and similar covariances. Large deviations from these assumptions may result in sub-optimal features.

Also the maximum rank of S_b in this formulation is $N_c - 1$ where N_c the number of different classes is. Thus basic LDA cannot produce more than $N_c - 1$ features. This is, however, simple to remedy by projecting the data onto a subspace orthogonal to the computed eigenvectors, and repeating the LDA analysis in this space.

However, the classification performance of traditional LDA is often degraded by the fact that their separability criteria are not directly related to their classification accuracy in the output space. A solution to the problem is to introduce weighting functions into LDA. Object classes that are closer together in the output space, and thus can potentially result in misclassification, should be more heavily weighted in the input space. This idea has been further extended in with the introduction of the fractional-step linear discriminant analysis algorithm (F-LDA), where the dimensionality reduction is implemented in a few small fractional steps allowing for the relevant distances to be more accurately weighted. Although the method has can be applied on low dimensional patterns it cannot be directly applied to high-dimensional patterns, such as those face images due to two factors: (1) the computational difficult of the eigen-decomposition of matrices in the high-dimensional image space; (2) the degenerated scatter matrices caused by the small sample size, which widely exists in the FR tasks where the number of training samples is smaller than the dimensionality of the samples.

The traditional solution to the SSS problem requires the incorporation of a PCA step into the LDA framework. In this approach, PCA is used as a pre-processing step for dimensionality reduction so as to discard the null space of the within-class scatter matrix of the training data set. Then LDA is performed in the lower dimensional PCA subspace. However, it has been shown that the discarded null space may contain significant discriminatory information. To prevent this from happening, solutions without a separate PCA step, called direct LDA (D-LDA) methods have been presented recently. In the D-LDA framework, data are processed directly in the original high-dimensional input space avoiding the loss of significant discriminatory information due to the PCA pre-processing step.

Firstly dimensionality of the original input space is lowered by introducing a new variant of D-LDA that results in a low-dimensional SSS-free subspace where the most discriminatory features are preserved. The variant of D-LDA utilizes a modified Fisher's criterion to avoid a problem resulting from the wage of the zero eigenvalues of the within-class scatter matrix as possible divisors. Also, a weighting function is introduced into the variant of D-LDA, so that a subsequent F-LDA step can be applied to carefully re-orient the SSS-free subspace resulting in a set of optimal discriminant features for face representation.

The DF-LDA is a linear pattern recognition method. Compared with nonlinear models, a linear model is rather robust against noises and most likely will not over fit. Although it has been shown that distribution of face patterns is highly non convex and complex in most cases, linear methods are still able to provide cost effective solutions to the FR tasks through integration with other strategies, such as the principle of "divide and conquer," in which a large and nonlinear problem is divided into a few smaller and local linear sub problems.

Let S_{BTW} and S_{WTH} denote the between- and within-class scatter matrices of the training image set, respectively. LDA-like approaches such as the Fisherface method find a set of basis vectors, denoted by that maximizes the ratio between S_{BTW} and S_{WTH} is

$$\Psi = \arg\max_{\Psi} \frac{\left| \left(\Psi^T S_{BTW} \Psi \right) \right|}{\left| \left(\Psi^T S_{WTH} \Psi \right) \right|} \tag{10}$$

The maximization process in (3) is not directly linked to the classification error which is the criterion of performance used to measure the success of the Face Recognition procedure. Thus, the weighted between-class scatter matrix can be expressed as:

$$S_{BTW} = \sum^{C} \Phi_i \Phi_i^T \tag{11}$$

where

$$\Phi_i = \left(L_i / L \right)^{1/2} \sum_{i=1}^{C} \left(w\left(d_{ij} \right) \right)^{1/2} \left(\overline{Z_i} - \overline{Z_j} \right), \overline{Z_i} \tag{12}$$

is the mean of class Z_i, L_i is the number of elements in Z_i, and

$$d_{i,j} = \left\| \overline{Z_i} - \overline{Z_j} \right\| \tag{13}$$

is the Euclidean distance between the means of class i and j .

Basis selection from DF-LDA

The set Y vectors are chosen by the equation $[y_1 \ldots . y_y] = \Pi_{<1..y>}(U^T S_{TOT} U^T)$ Where S_{TOT} is the sum of between and within class scatter matrices, U is a diagonal matrix from Eigen values and vectors. Fisher faces are shown in figure 7 below.



Fig. 7. Fisher Faces from DF-LDA

We can clearly see from fisher faces that more pronounced features are highlighted than the rest of the face point like hair, eyebrows etc.

6. Locality preserving projections

Different from Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) which effectively see only the Euclidean structure of face space, LPP finds an embedding that preserves local information, and obtains a face subspace that best detects the essential face manifold structure. The Laplacianfaces are the optimal linear approximations to the eigen functions of the Laplace Beltrami operator on the face manifold. In this way, the unwanted variations resulting from changes in lighting, facial expression, and pose may be eliminated or reduced. Theoretical analysis shows that PCA, LDA and LPP can be obtained from different graph models. By using Locality Preserving Projections (LPP), the face images are mapped into a face subspace for analysis.

LPP shares many of the data representation properties of nonlinear techniques such as Laplacian Eigen maps or Locally Linear Embedding. Yet LPP is linear and more crucially is

defined everywhere in ambient space rather than just on the training data points. It builds a graph incorporating neighborhood information of the data set. Using the notion of the Laplacian of the graph, transformation matrix is computed which maps the data points to a subspace.

This linear transformation optimally preserves local neighborhood information in a certain sense. The representation map generated by the algorithm may be viewed as a linear discrete approximation to a continuous map that naturally arises from the geometry of the manifold. In the meantime, there has been some interest in the problem of developing low dimensional representations through kernel based techniques for face recognition. These methods can discover the nonlinear structure of the face images. However, they are computationally expensive. Moreover, none of them explicitly considers the structure of the manifold on which the face images possibly reside.

While the Eigen faces method aims to preserve the global structure of the image space, and the Fisher faces method aims to preserve the discriminating information; our Laplacianfaces method aims to preserve the local structure of the image space. In many real world classification problems, the local manifold structure is more important than the global Euclidean structure, especially when nearest neighbor like classifiers are used for classification.

LPP seems to have discriminating power although it is unsupervised. An efficient subspace learning algorithm for face recognition should be able to discover the nonlinear manifold structure of the face space. LPP shares some similar properties to LLE, such as a locality preserving character. However, their objective functions are totally different. LPP is obtained by finding the optimal linear approximations to the eigen functions of the Laplace Beltrami operator on the manifold. LPP is linear, while LLE is nonlinear. Moreover, LPP is defined everywhere, while LLE is defined only on the training data points and it is unclear how to evaluate the maps for new test points. In contrast, LPP may be simply applied to any new data point to locate it in the reduced representation space. LPP seeks to preserve the intrinsic geometry of the data and local structure.

The objective function of LPP is as follows:

$$\min \sum_{ij} \left(y_i - y_j \right)^2 S_{ij} \tag{14}$$

Where

$$S_{ij} = \begin{cases} \exp\left(\left\| x_i - x_j \right\|^2 / t \right), & \left\| x_i - x_j \right\|^2 < \varepsilon \\ 0 \end{cases} \tag{15}$$

7. Statistical view of LPP

LPP can also be obtained from statistical viewpoint. Suppose the data points follow some underlying distribution. Let d be the number of non-zero S_{ij}, and D be a diagonal matrix whose entries are column (or row, since S is symmetric) sums of S, $D_{ii}=\sum_j S_{ji.}$ By the Strong Law of Large Numbers, $E(zz^T \mid ||z|| < \varepsilon)$ can be estimated from the sample points as follows:

$$E\left(ZZ^T \mid \|z\|<\varepsilon\right)$$

$$\approx \frac{1}{d}\sum_{\|Z\|<\varepsilon} ZZ^T$$

$$= \frac{1}{d}\sum_{\|x_i - x_j\|<\varepsilon} \left(x_i - x_j\right)\left(x_i - x_j\right)^T$$

$$= \frac{1}{d}\sum_{i,j} \left(x_i - x_j\right)\left(x_i - x_j\right)^T S_{ij}$$

$$= \frac{1}{d}\left(\sum_{i,j} x_i x_i^T S_{ij} + \sum_{i,j} x_j x_j^T S_{ij} - \sum_{i,j} x_i x_j^T S_{ij} - \sum_{i,j} x_j x_i^T S_{ij}\right)$$

$$= \frac{2}{d}\left(\sum_{i,j} x_i x_i^T D_{ii} - \sum_{i,j} x_i x_j^T S_{ij}\right)$$

$$= \frac{2}{d}\left(XDX^T - XSX^T\right)$$

$$= \frac{2}{d}XLX^T \tag{16}$$

where $L = D - S$ is the Laplacian matrix. The *ith* column of matrix X is x*i*.

8. Theoretical analysis of LPP, PCA AND LDA

In this section, we present a theoretical analysis of LPP and its connections to PCA and LDA.

8.1 Connections to PCA

It is worthwhile to point out that XLX^T is the data covariance matrix, if the Laplacian matrix L *is*

$$\frac{1}{n}I - \frac{1}{n^2}ee^T \tag{17}$$

where n is the number of data points, I is the identity matrix and e is a column vector taking 1 at each entry. In fact, the Laplacian matrix here has the effect of removing the sample mean from the sample vectors.

In this case, the weight matrix S takes $1/n^2$ at each entry, i.e

$$S_{ij} = 1/n^2, \forall i, j \tag{18}$$

$$D_{ii} = \sum_j S_{ji} = 1/n \tag{19}$$

Hence the Laplacian matrix is

$$L = D - S = \frac{1}{n} I - \frac{1}{n^2} ee^T \tag{20}$$

Let m denote the sample mean i.e.

$$m = 1 / n \sum_i x_i \tag{21}$$

we have

$$XLX^T = \frac{1}{n} X \left(I - \frac{1}{n^2} ee^T \right) X^T$$

$$= \frac{1}{n} XX^T - \frac{1}{n^2} (X_e)(X_e)^T$$

$$= \frac{1}{n} \sum_i x_i x_i^T - \frac{1}{n^2} (nm)(nm)^T$$

$$= \frac{1}{n} \sum_i (x_i - m)(x_i - m)^T + \frac{1}{n} \sum_i x_i m^T + \frac{1}{n} \sum_i m x_i^T - \frac{1}{n} \sum_i mm^T - mm^T$$

$$= E[(x - m)(x - m)]^T + 2mm^T - 2mm^T$$

$$= E[(x - m)(x - m)]^T \tag{22}$$

Where $E[(x - m)(x - m)]^T$, is just the covariance matrix of the data set

The above analysis shows that the weight matrix S plays a key role in the LPP algorithm. When we aim at preserving the global structure, we take ε (or k) to be infinity and choose the eigenvectors (of the matrix XLX^T) associated with the largest eigenvalues. Hence the data points are projected along the directions of maximal variance. ε should be sufficiently small to preserve the local structure and choose the Eigen vectors associates with smallest Eigen values.

Hence the data points are projected along the directions preserving locality. It is important to note that, when ε (or k) is sufficiently small, the Laplacian matrix is no longer the data covariance matrix, and hence the directions preserving locality are not the directions of minimal variance. In fact, the directions preserving locality are those minimizing *local* variance.

8.2 Connections to LDA

LDA seeks directions that are efficient for discrimination. The projection is found by solving the generalized Eigen value problem

$$S_B w = \lambda S_W w \tag{23}$$

where S_B and S_W are between and within class scatter matrices. Suppose there are *l* classes. The ith class contains *ni* sample points. Let m(*i*) denote the average vector of the *i*th class. Let x(*i*) denote the random vector associated tothe *i*th class and) (i *j* x denote the *j*th sample point in the *i*th class. We can rewrite the matrix S_w as follows:

$$S_W = \sum_{i=1}^{l} \left(\sum_{j=1}^{n_i} \left(x_j^{(i)} - m^{(i)} \right) \left(x_j^{(i)} - m^{(i)} \right)^T \right)$$

$$= \sum_{i=1}^{l} \left(\sum_{j=1}^{n_i} \left(x_j^{(i)} \left(x_j^{(i)} \right)^T - m^{(i)} \left(m^{(i)} \right)^T - x_j^{(i)} \left(m^{(i)} \right)^T + m^{(i)} \left(m^{(i)} \right)^T \right) \right)$$

$$= \sum_{i=1}^{l} \left(\sum_{j=1}^{n_i} \left(x_j^{(i)} \left(x_j^{(i)} \right)^T - n_i m^{(i)} \left(m^{(i)} \right)^T \right) \right)$$

$$= \sum_{i=1}^{l} \left(X_i X_i^T - \frac{1}{n_i} \left(x_1^{(i)} + \ldots + x_{ni}^{(i)} \right) \left(x_1^{(i)} + \ldots + x_{ni}^{(i)} \right)^T \right)$$

$$= \sum_{i=1}^{l} \left(X_i X_i^T - \frac{1}{n_i} X_i \left(e_i e_i^T \right) X_i^T \right)$$

$$= \sum_{i=1}^{l} X_i L_i X_i^T \tag{24}$$

Where,

$X_i L_i X_i^T$ is the data covariance matrix of the ith class and

X_i = [$X_1^{(i)}$, $X_2^{(i)}$, $X_3^{(i)}$,…. $X_{ni}^{(i)}$] is a d x n_i matrix.

$L_i = I - 1 / n_i e_i e_i^T$ is a n_i x n_i matrix where I is the identity matrix and $e_i (1,1,1....1)^T$ is an *ni* dimensional vector.

To further simplify the above equation, we define

$$W_{ij} = \begin{cases} 1 / n_k & \text{if } x_i \text{ and } x_j \text{ both belong to the kth class} \\ 0 & \text{otherwise,} \end{cases} \tag{25}$$

It is interesting to note that we could regard the matrix *W* as the weight matrix of a graph with data points as its nodes. Specifically, *Wij* is the weight of the edge (x*i*, x*j*). *W* reflects the class relationships of the data points. The matrix *L* is thus called *graph Laplacian*, which plays key role in LPP.

Similarly, we can compute the matrix *SB* as follows:

$$S_B = \sum_{i=1}^{l} n_i \left(m^{(i)} - m \right) \left(m^{(i)} - m \right)^T$$

$$= \left(\sum_{i=1}^{l} n_i m^{(i)} \left(m^{(i)} \right)^T \right) - m \left(\sum_{i=1}^{l} n_i \left(m^{(i)} \right)^T \right) - \left(\sum_{i=1}^{l} n_i m^{(i)} \right) m^T + \left(\sum_{i=1}^{l} n_i \right) m m^T$$

$$= \sum_{i=1}^{l} \left(\frac{1}{n_i} \left(x_1^{(i)} + \ldots + x_{ni}^{(i)} \right) \left(x_1^{(i)} + \ldots + x_{ni}^{(i)} \right)^T \right) - 2nmm^T + nmm^T$$

$$= \left(\sum_{i=1}^{l} \sum_{j,k=1}^{n_i} \frac{1}{n_i} x_j^{(i)} \left(x_j^{(i)} \right)^T \right) - 2nmm^T + nmm^T$$

$$= XWX^T - 2nmm^T + nmm^T$$

$$= XWX^T - nmm^T$$

$$= XWX^T - X \left(\frac{1}{n} e e^T \right) X^T$$

$$= X \left(W - \frac{1}{n} e e^T \right) X^T$$

$$= X \left(W - I + I - \frac{1}{n} e e^T \right) X^T$$

$$= -XLX^T + X \left(I - \frac{1}{n} e e^T \right) X^T$$

$$= -XLX^T + C \tag{26}$$

where e = $(1,1,\ldots,1)^T$ is a n dimensional vector and $C = X \left(I - \frac{1}{n} e e^T \right) X^T$ is the data covariance matrix.

Thus, the generalized eigenvector problem of LDA can be written as follows:

$$S_B w = \lambda S_W w \tag{27}$$

$$\Rightarrow \left(C - XLX^T \right) w = \lambda XLX^T w$$

$$\Rightarrow Cw = \left(1 + \lambda \right) XLX^T w$$

$$\Rightarrow XLX^T w = \frac{1}{1+\lambda} Cw$$

Thus, the projections of LDA can be obtained by solving the following generalized eigenvalue problem,

$$\Rightarrow XLX^T w = \lambda Cw \tag{28}$$

The optimal projections correspond to the eigenvectors associated with the smallest eigenvalues. If the sample mean of the data set is zero, the covariance matrix is simply XX^T which is close to the matrix XDX^T in the LPP algorithm. Our analysis shows that LDA actually aims to preserve discriminating information and global geometrical structure. Moreover, LDA has a similar form to LPP. However, LDA is supervised while LPP can be performed in either supervised or unsupervised manner.

8.3 Learning laplacian faces for representation

LPP is a general method for manifold learning. It is obtained by finding the optimal linear approximations to the eigenfunctions of the Laplace Betrami operator on the manifold. Therefore, though it is still a linear technique, it seems to recover important aspects of the intrinsic nonlinear manifold structure by preserving local structure. Based on LPP, Laplacianfaces method for face representation is a locality preserving subspace. In the face analysis and recognition problem one is confronted with the difficulty that the matrix XDX^T is sometimes singular. This stems from the fact that sometimes the number of images in the training set (n) is much smaller than the number of pixels in each image (m). In such a case, the rank of XDX^T is at most n, while XDX^T is an $m \times m$ matrix, which implies that XDX^T is singular. To overcome the complication of a singular XDX^T, we first project the image set to a PCA subspace so that the resulting matrix XDX^T is nonsingular. Another consideration of using PCA as preprocessing is for noise reduction. This method, we call *Laplacianfaces*, can learn an optimal subspace for face representation and recognition.

The algorithmic procedure of Laplacianfaces is formally stated below:

1. PCA projection: We project the image set {x*i*} into the PCA subspace by throwing away the smallest principal components.
2. Constructing the nearest-neighbor graph: Let G denote a graph with n nodes. The *ith* node corresponds to the face image x_i. We put an edge between nodes i and j if x_i and x_j are "close", i.e. x*i* is among k nearest neighbors of x*i* or x*i* is among k nearest neighbors of x_j. The constructed nearest neighbor graph is an approximation of the local manifold structure. Note that, here we do not use the ε - neighborhood to construct the graph. This is simply because it is often difficult to choose the optimal ε in the real world applications, while k nearest neighbor graph can be constructed more stably. The disadvantage is that the k nearest neighbor search will increase the computational complexity of our algorithm. When the computational complexity is a major concern, one can switch to the ε -neighborhood.
3. Choosing the weights: If node i and j are connected, put

$$S_{ij} = e^{\frac{\left\| x_i - x_j \right\|^2}{t}} \tag{29}$$

where t is a suitable constant. Otherwise, put $S_{ij} = 0$. The weight matrix S of graph G models the face manifold structure by preserving local structure.

4. Eigenmap: Compute the eigenvectors and eigenvalues for the generalized eigenvector problem:

$$XLX^T w = \lambda XDX^T w \tag{30}$$

where D is a *k*-dimensional vector. *W* is the transformation matrix. This linear mapping best preserves the manifold's estimated intrinsic geometry in a linear sense. The column vectors of *W* are the so called *Laplacianfaces*.

9. Face representation using laplacianfaces

As we described previously, a face image can be represented as a point in image space. A typical image of size *m*×*n* describes a point in *m*×*n*-dimensional image space. However, due to the unwanted variations resulting from changes in lighting, facial expression, and pose, the image space might not be an optimal space for visual representation, we have discussed how to learn a locality preserving face subspace which is insensitive to outlier and noise. The images of faces in the training set are used to learn such a locality preserving subspace. The subspace is spanned by a set of eigenvectors of equation (1), i.e. w0, w1, …, wk-1.

Eigenmaps are obtained from the generalized eigenvector problem as $ALA^T a = \lambda ADA^T a$ where D is a diagonal matrix whose entries are column or row, since W is symmetric sums of W, $D_{ii} = \Sigma j W_{ji}$., L = D -W is the Laplacian matrix is equivalent nonlinear Laplace Beltrami opearator. The ith column of matrix A is xi. Let the column vectors a_0; _ _ _ ; a_{l-1} be the solutions of equation (), ordered according to their eigenvalues, in ascending order Thus, the embedding is as follows: $yi = E^T x_i$; E = (a_0; a_1; _ _ _ ; a_{l-1}) where y_i is a l-dimensional vector, and E is a n x l matrix.. The y_i represent the Laplacian faces.

9.1 Basis selection from LPP

Locality information can be preserved by the following transformation on A, the input face space [z_1 …. Z_z] = $\Pi_{<1..z}(A^T L A)$ Where L =D-W gives the Laplacian matrix. D is the diagonal matrix and W is the weight matrix of the K nearest neighbors clustering.

Basis for the face space is obtained as, $B = [X_1\ X_2\ ....X_x\ Y_1\ Y_2.....Y_y\ Z_1\ Z_2.....Z_z]$, such that

$$x + y + z = \frac{M}{3} \tag{31}$$

and

$$x + y + z = \frac{2M}{3} \tag{32}$$

where M is the dimension of the original face space

9.2 Projection onto reduced subspace

Each face in the training set Φ_i can be represented as a linear combination of these vectors, U_i ε B , $1 \le i \le K$ such that $\Phi_i = \sum_{j=1}^{k} w_j u_j$, where u_j's are Eigenfaces. These weights are calculated as: $w_j = u_j^T \Phi_i \Omega_i = [w_1\ w_2 ... w_k]$ i.e. the orthogonal projection of a face vector on each basis vector.

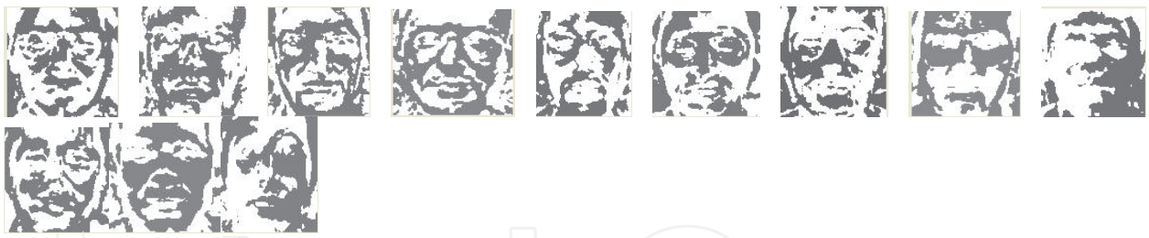


Fig. 8. Laplacian Faces from LPP

10. Independent component analysis

Independent component analysis (ICA) is a statistical method, the goal of which is to decompose multivariate data into a linear sum of non-orthogonal basis vectors with coefficients (encoding variables, latent variables, hidden variables) being statistically independent.

ICA generalizes a widely-used subspace analysis method such as principal component analysis (PCA) and factor analysis, allowing latent variables to be non-Gaussian and basis vectors to be non-orthogonal in general. ICA is a density estimation method where a linear model is learned such that the probability distribution of the observed data is best captured, while factor analysis aims at best modeling the covariance structure of the observed data.

The ICA model is a generative model, which means that it describes how the observed data are generated by a process of mixing the components *si*. The independent components are latent variables, meaning that they cannot be directly observed. Also the mixing matrix is assumed to be unknown. All we observe is the random vector X, and we must estimate both A and S using it. This must be done under as general assumptions as possible. The starting point for ICA is the very simple assumption that the components S_i are statistically *independent*. It will be seen below that we must also assume that the independent component must have *nongaussian* distributions. However, in the basic model we do *not* assume these distributions known (if they are known, the problem is considerably simplified.) For simplicity, we are also assuming that the unknown mixing matrix is square, but this assumption can be sometimes relaxed. Then, after estimating the matrix A, we can compute its inverse, say W, and obtain the independent component simply by: s=Wx

ICA is very closely related to the method called *blind source separation* (BSS) or blind signal separation. A "source" means here an original signal, i.e. independent component, like the speaker in a cocktail party problem. "Blind" means that we know very little, if anything, on the mixing matrix, and make little assumptions on the source signals. ICA is one method, perhaps the most widely used, for performing blind source separation.

The task of ICA is to estimate the mixing matrix A or its inverse $W = A^{-1}$ such that elements of the estimate $y = A^{-1}x = Wx$ are as independent as possible. For the sake of simplicity, we often leave out the index t if the time structure does not have to be considered.

PCA makes one important assumption: the probability distribution of input data must be Gaussian. When this assumption holds, covariance matrix contains all the information of (zero-mean) variables. Basically, PCA is only concerned with second-order (variance) statistics. The mentioned assumption need not be true. If we presume that face images have

more general distribution of probability density functions along each dimension, the representation problem has more degrees of freedom. In that case PCA would fail because the largest variances would not correspond to meaningful axes of PCA.

$$x_i(t) = a_i1*s_1(t) + a_i2*s_2(t) + a_i3*s_3(t) + a_i4*s_4(t) \ldots \quad (33)$$

Here, i =1:4.
In vector-matrix notation, and dropping index t, this is

$$x = A\, s \quad (34)$$

$$s = A^{-1}\, x \quad (35)$$

$$s = W\, x \quad (36)$$

$$W = A^{-1} \quad (37)$$

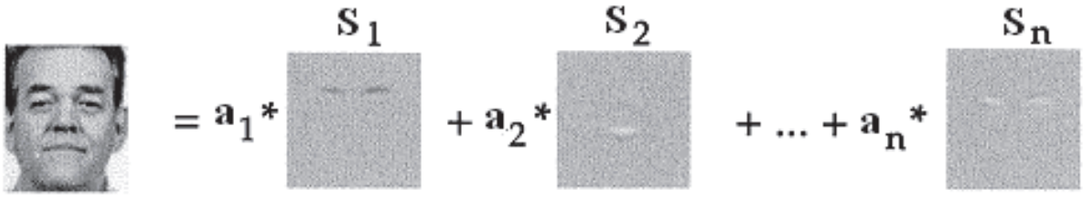


Fig. 9. Mixture Matrix forming face

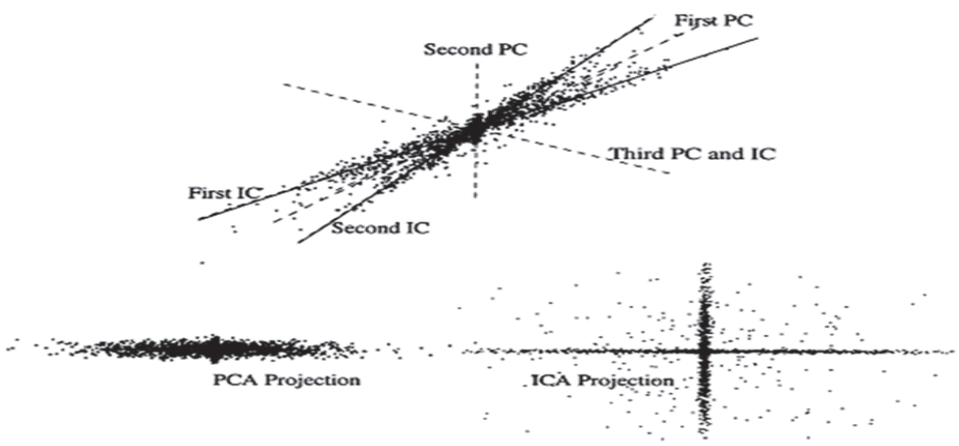


Fig. 10. Different Principal Component(PC) directions & PCA vs. ICA Projections

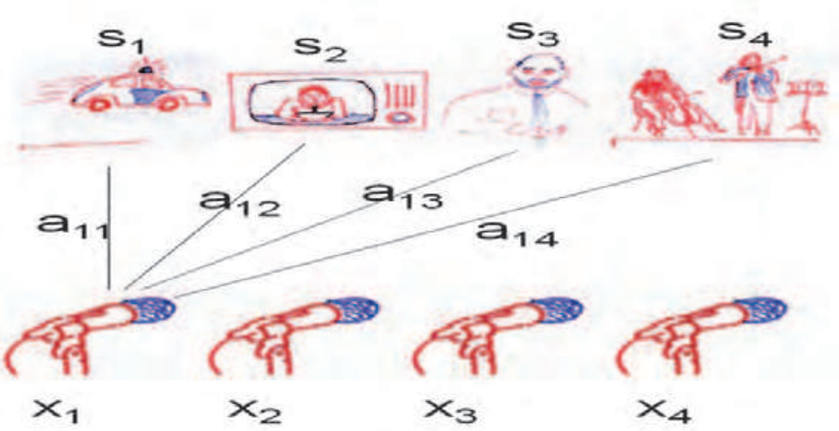


Fig. 11. x=As (Blind Source Separation)

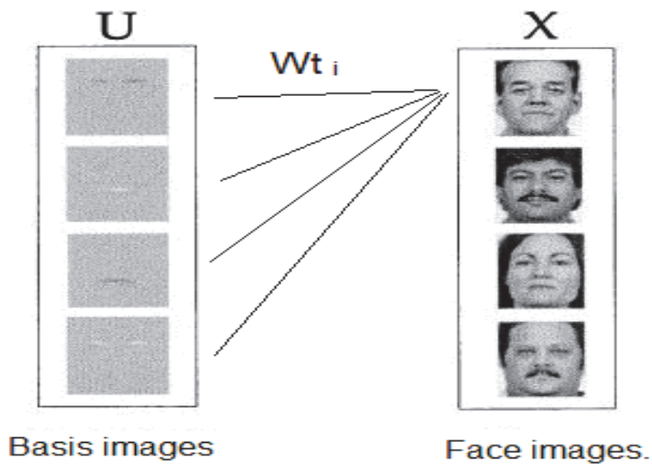


Fig. 12. Construction of face from Basis



Fig. 13. Basis Images from ICA

11. Random projections

There has been a strong trend lately in face processing research away from geometric models towards appearance models. Appearance-based methods employ dimensionality reduction to represent faces more compactly in a low-dimensional subspace which is found by optimizing certain criteria. Recently, Random Projection (RP) has emerged as a powerful

method for dimensionality reduction. It represents a computationally simple and efficient method that preserves the structure of the data without introducing significant distortion. D $O(\log n / \varepsilon^2)$ dimensional subspace such that the distances between the points are approximately preserved.

Transforms Γ_i to a lower dimension *d*, with d<<p via the following transformation: $\Gamma_i = R\Gamma_i$ where *R* is orthonormal and its columns are realizations of independent and identically distributed (i.i.d.) zero-mean normal variables, scaled to have unit length. RP is motivated by the *Johnson-Lindenstrauss* lemma that states that a set of *M* points in a high dimensional.

Euclidean space can be mapped down onto a $d \geq O(\log n / \varepsilon^2)$ dimensional subspace such that the distances between the points are approximately preserved.

The main reason for orthogonalizing the random vectors is to preserve the similarities between the original vectors in the low-dimensional space. In high enough dimensions, however, it is possible to save computation time by avoiding the orthogonalization step without affecting much the quality of the projection matrix. This is due to the fact that, in high-dimensional spaces, there exist a much larger number of almost orthogonal vectors than orthogonal vectors. Thus, high-dimensional vectors having random directions are very likely to be close to orthogonal.

12. Mixture of components

One can use different ratios of feature vectors drawn from SVD, DF-LDA & LPP Techniques. The first step can be normalizing the images in the training set to compensate for the illumination effects. These processed images should be subjected to dimensionality reduction using each of the methods mentioned in the chapter. Basis selection can be carried out using these independent sets of dimension reduced vectors in different proportions aimed at enhancing the efficiency and accuracy of recognition task. Below is a sample example mentioned with two trials, one with for 1/3rd dimensionality reduction and another with 2/3rd reduction. In each of the trials, several iterations are performed by taking different combinations of the feature vectors. The iterations will converge when the desired precision of recognition rate is obtained.

12.1 Example

12.1.1 Preprocessing

The Face Space: For the recognition task , each m X n I_i image in the training set is transformed into a column vector of mn components. A matrix S (mn X M) is constructed such that S =[$I_1 I_2 \ldots I_M$] , where M is number of face images in training set It is found that all N vectors are linearly independent, which implies that the range space of matrix S is the entire region spanned by the columns of S. i.e Range space of S R(S)=[S]

Normalization: Normalize the images ,to reduce illumination effects and lighting conditions as,

$$Ai = (\Phi - \mu_i)X \frac{\delta'}{\delta} + \mu' \tag{40}$$

For i=1,2,3….., M
Where,

$$\mu_i = \frac{1}{N^2}\sum_{j=1}^{N^2} x_j \tag{41}$$

$$\mu'_i = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{N^2}\sum_{j=1}^{N^2} A_{ij} \tag{42}$$

$$\delta_i = \sqrt[2]{\frac{1}{N^2-1}\sum_{j=1}^{N^2}\left(x_j - \mu_i\right)} \tag{43}$$

$$\delta'_i = \frac{1}{N^2}\sum_{i=1}^{N^2}\sqrt[2]{\frac{1}{N^2-1}\sum_{j=1}^{N^2} A_{ji}} \tag{44}$$

12.1.2 Basis selection

Recognition Task: Unknow probeface is normalized (ϕ) and projected on to the subspace to get weight for the probe image $w_i = u_j^T \Phi$ Euclidean Distance measure is used in classification given by $e_r = \min \|\Omega - \Omega_i\|$. And if $e_r < \Theta$ where Θ is a threshold chosen heuristically, then we the probe image is recognized as the image with which it gives the lowest score. If however $e_r > \Theta$ then the probe does not belong to the database.

Deciding on the Threshold: A set of 150 known images other than the ones in the data set is used in the computation of threshold θ given by $\theta = \mu + \eta\sigma$. Where,

$$\mu = \frac{1}{N}\sum_{i=0}^{N-1} x_i \tag{45}$$

$$\sigma^2 = \frac{1}{N-1}\sum_{i=0}^{N-1}\left(x_j - \mu\right)^2 \tag{46}$$

$\eta \in I$ is chosen according to level of precision required in the results. $x_i \in \gamma$

The method of choosing right combination of right proportion of feature vectors has been applied on a large database consisting of a variety of still images with illumination, expression variations as well as partially occluded images. The ratio 3:2:5:: SVD:DF-LDA:LPP has yielded highest accuracy in recognition. The example is tried on a total test set of 165 images drawn from YALE dataset and the training set consisting 15 classes having a class count of five images.

An ROC graph is plotted to visualize and analyze the working of face recognition efficiency. It is a two dimensional graph in which TP rate, true positive rate, is plotted on the Y axis and FP rate, false positive rate, is plotted on the X axis. Given a set of test images a two by two contingency table is constructed representing the dispositions of the set of images.

SVD (no. of vectors)	DFLDA (no. of vectors)	LPP (no. of vectors)	EFFICIENCY (in %)
15	5	5	80.00
5	5	15	81.21
8	9	8	81.81
15	5	15	87.27
5	15	5	81.21

Table 1. Iterations for subspace of dimension M/3

Graph No.	True Positive	False Negative	False Positive	True Negative
1	122	28	5	10
2	123	27	4	11
3	125	15	5	10
4	132	18	3	12
5	122	28	3	12

Table 2. Comparative results with Iteration Trial of M/3

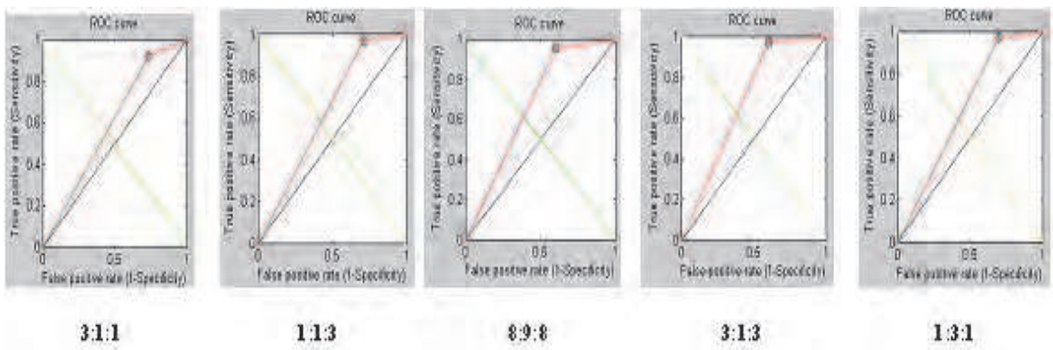


Fig. 14. ROC's Indicating the True Positive VS False Positive for M/3

SVD (no. of vectors)	DFLDA (no. of vectors)	LPP (no. of vectors)	EFFICIENCY (in %)
30	10	10	84.24
10	10	30	85.45
20	15	15	86.67
25	15	10	84.84
15	10	25	92.12

Table 3. Iterations for subspace of dimension 2M/3

Graph No.	True Positive	False Negative	False Positive	True Negative
1	129	21	5	10
2	132	18	4	11
3	133	17	5	10
4	128	32	4	11
5	140	10	3	12

Table 4. Comparative results with Iterartion Trial of M/3

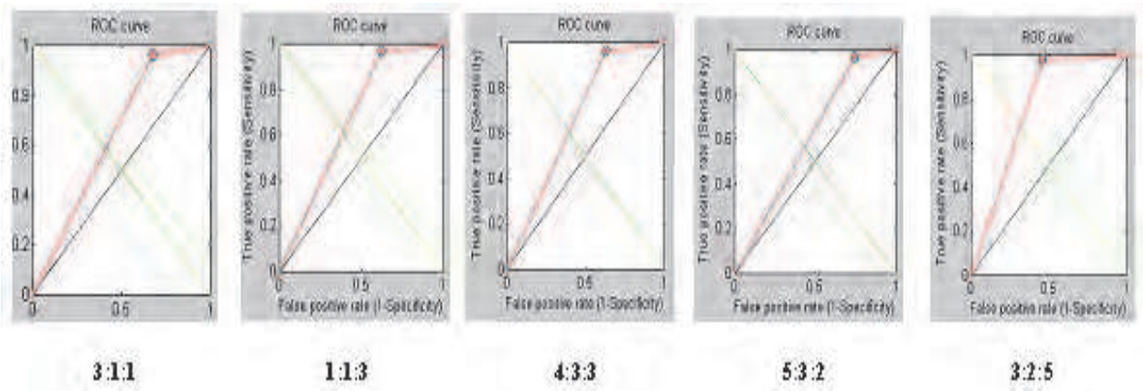


Fig. 15. ROC's Indicating the True Positive VS False Positive for 2M/3

13. Summary

In this chapter several linear and non linear dimensionality reduction techniques were discussed from the perspective of face recognition. Since the face images contain several characteristic features both global and local, using any one method alone may not yield better recognition accuracy. It may be good to have combinations of the basis vectors from several approaches to achieve higher accuracy. Underlying manifold structure in image space will get face subspace and is possible with LPP, ICA methods. More pronounced features can be drawn from the space in case of LDA based algorithms. Random and PCA projections give appearance models which are holistic in nature.

Future of face recognition can also look at increase in dimension like depth information for recognition purposes. Algorithmic models should aim at addressing scale invariance feature vectors which can hopefully solve recognition task even under extreme variations in images.

The approach to face recognition was motivated by information theory, leading to the idea of basing face recognition on a small set of image features that best approximate the set of known face images, without requiring that they correspond to our intuitive notions of facial parts and features. The approach does provide a practical solution to the problem of face recognition and is relatively simple and has been shown that it can work well in a

constrained environment. Anecdotal experimentation with acquired image sets indicates that profile size, complexion, ambient lighting and facial angle play significant parts in the recognition of a particular image.

14. References

R.Hietmeyer.Biometric identification promises fast and secure processing of airline passengers.The International Civil Aviation Organization Journal ,55(9):10 – 11,2000.

Michael E. Wall, Andreas Rechtsteiner, Luis M. Rocha, "Singular Value Decomposition And Principal Component Analysis" , A Practical Approach to Microarray Data Analysis , Chapter 5, pp. 91-109, 2003

M.A. Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuro-science, vol. 3, pp.71-86, 1991.

Xiaofei He, Partha Niyogi, " Locality Preserving Projections", IN the proceedings of Advances in Neural Information Processing Systems, 2003.

Juwei Lu, K.N. Plataniotis, and A.N. Venetsanopoulos, "Face Recognition Using LDA Based Algorithms", IEEE Transactions on Neural Networks, VOL. 14, NO. 1,pp. 195-200, 2003.

Yu, H., Yang, J.,"A Direct LDA #algorithm for High-Dimensional Data with Application to Face Recognition. Pattern Recognition, Vol.34, pp.2067–2070., 2001

Sam Roweis, and Lawrence K. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," In the proceedings of Science Journal, vol 290, pp. 2323—2326, 2000.

Joshua B. Tenenbaum,* Vin de Silva,John C. Langford ,"A Global Geometric Framework for Nonlinear Dimensionality Reduction" , In the proceedings of Science Journal, vol 290, pp. 2319-2323, 2000.

Xiaofei He, Shuicheng Yan, Yuxiao Hu, Partha Niyogi, and Hong-Jiang Zhang, "Face Recognition Using Laplacianfaces", In the proceedings of IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 3, pp. 328- 340, 2005

Berry, M. W., Dumais, S. T., and O'Brian, G. W.(1995). Using Linear Algebra for Intelligent Information Retrieval., In the proceedings of SIAM Review Journal, 37(4).pp. 573—595, 1995.

J. Liu and S. Chen. Discriminant common vecotors versus neighbourhood components analysis and laplacianfaces: A comparative study in small sample size problem. Image and Vision Computing, 24(3):249–262, 2006.

Navin Goel*a*, George Bebis*a*, Ara Nefian*b* "Face Recognition Experiments with Random Projection" ,

Muhammad Imran Razzak, Muhammad Khurram Khan, Khaled Alghathbar, Rubiyah Yousaf, "Face Recognition using Layered Linear Discriminant Analysis and Small Subspace", In the 10th IEEE International Conference on Computer and Information Technology (CIT 2010), pp. 1407-1412, 2010.

MaxWelling, "Fisher Linear Discriminant Analysis", Classnotes in Machine Learning.

Kari Torkkola, "Discriminative Features for Text Document Classification", In the proceedings of International Conference on Pattern Recognition, pp. 472-475, 2002.

Shermina.J, "Application of Locality Preserving Projections in Face Recognition", In the proceedings of International Journal of Advanced Computer Science and Applications, Vol. 1, No. 3, pp. 82 -85 September 2010

Deng Cai, Xiaofei He, Jiawei Han, Hong-Jiang Zhang," Orthogonal Laplacianfaces for Face Recognition ", In the proceedings of IEEE Transactions on Image Processing,pp.1-7 2010

S.Sakthivel, R.Lakshmipathi, "Enhancing Face Recognition Using Improved Dimensionality Reduction And Feature Extraction Algorithms –An Evaluation With ORL Database", International Journal of Engineering Science and Technology, Vol. 2(6), pp.2288-2295, 2010.

Ruba Soundar Kathavarayan,, Murugesan Karuppasamy, "Preserving Global and Local Features for Robust Face Recognition under Various Noisy Environments", In the proceedings of International Journal of Image Processing (IJIP) Volume(3), Issue(6), pp. 328-340, 2010.

Neeta Nain, Prashant Gour, Nitish Agarwal, Rakesh P Talawar, Subhash Chandra, "Face Recognition using PCA and LDA with Singular Value Decomposition(SVD) using 2DLDA", Proceedings of the World Congress on Engineering Vol I, ISBN:978-988-98671-9-5, pp. 1-4, 2008.

Tat-Jun Chin, Konrad Schindler, David Suter, "Incremental Kernel SVD for Face Recognition with Image Sets", In the proceedings of Seventh IEEE International Conference on Automatic Face and Gesture Recognition pp.461-466, 2006.

Marian Stewart Bartlett, , Javier R. Movellan, Terrence J. Sejnowski, "Face Recognition by Independent Component Analysis", IEEE Transactions on Neural Networks, Vol. 13, NO. 6, , pp. 1450-1464, 2002.

K.J. Karande, S.N. Talbar, 'Independent Component Analysis of Edge Information for Face Recognition', In the Proceedings of International Journal of Image Processing vol.3, issue 3, 120-130, 2009.

M. Belkin, P. Niyogi, "Using Manifold Structure for Partially Labeled Classification", In the Proceedings of Conference on Advances in Neural Information Processing System, 2002.

W. Zhao, R. Chellappa, P.J. Phillips, 'Subspace Linear Discriminant Analysis for Face Recognition', Technical Report CAR-TR-914, Center for Automation Research, University of Maryland, 1999.

P. C. Yuen and J. H. Lai, "Independent Component Analysis of Face Images,", In the proceedings of IEEE Workshop Biologically Motivated Computer Vision, Seoul, Korea, 2000.

A. M. Martinez and A. C. Kak, "PCA versus LDA," In the proceedings of IEEE Transaction on Pattern Analysis and Machine Intelligence,, vol. 23, no. 2, pp. 228–233, 2001.

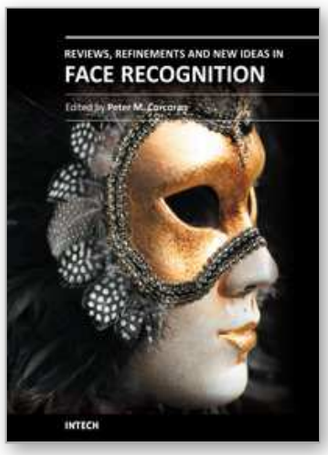
T. Shakunaga and K. Shigenari, "Decomposed Eigenface for Face Recognition under Various Lighting Conditions," In the proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2001.

Zhonglong Zheng, Fan Yang, Wenan Tan, Jiong Jia and Jie Yang, "Gabor feature-based face recognition using supervised locality preserving projection", In the proceedings of *Signal Processing,* Vol. 87, Issue 10, pp. 2473-2483, 2007.

Books
Santosh S. Vempala ,"The Random Projection Method",DIMACS 5th Edition
Rafael C. Gonzalez, Richard E. Woods,"Digital Image Processing" 2nd Edition

IntechOpen

IntechOpen



Reviews, Refinements and New Ideas in Face Recognition

Edited by Dr. Peter Corcoran

ISBN 978-953-307-368-2
Hard cover, 328 pages
Publisher InTech
Published online 27, July, 2011
Published in print edition July, 2011

As a baby one of our earliest stimuli is that of human faces. We rapidly learn to identify, characterize and eventually distinguish those who are near and dear to us. We accept face recognition later as an everyday ability. We realize the complexity of the underlying problem only when we attempt to duplicate this skill in a computer vision system. This book is arranged around a number of clustered themes covering different aspects of face recognition. The first section on Statistical Face Models and Classifiers presents reviews and refinements of some well-known statistical models. The next section presents two articles exploring the use of Infrared imaging techniques and is followed by few articles devoted to refinements of classical methods. New approaches to improve the robustness of face analysis techniques are followed by two articles dealing with real-time challenges in video sequences. A final article explores human perceptual issues of face recognition.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Shylaja S. Sharath, Balasubramanya Murthy K N and Natarajan S (2011). Dimensionality Reduction Techniques for Face Recognition, Reviews, Refinements and New Ideas in Face Recognition, Dr. Peter Corcoran (Ed.), ISBN: 978-953-307-368-2, InTech, Available from: http://www.intechopen.com/books/reviews-refinements-and-new-ideas-in-face-recognition/dimensionality-reduction-techniques-for-face-recognition

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License, which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen