INTRODUCTION

1. **Abstract**
2. **User requirements**
3. **State of the art**
4. **Facial Recognition**

The challenge of face recognition can be formulated as followed : with one or several images of a face, the goal would be to find or check the identity of a person by comparing his face to all the face images stored in a database. It is the most common and even popular skill. By the way this skill remains the most acceptable because it more suits with what human beings use in visual interaction; and compared to other methods, the face recognition seems more advantageous, in fact it is a non-intrusive method, in other words it does not require the cooperation of the subject, and a moreover the sensors used are cheaper.

* 1. **Facial recognition process**

Any facial recognition process must take into consideration several factors that contribute to the complexity of its task, because a face is a dynamic entity which constantly changes under the influence of several factors. A facial recognition system is generally divided into the following steps (see the figure):

Humans faces

Image capture

Extraction characters

Statistical analysis

Learning

Decision and classification

Facial recognition is facing the following problems:

* Change of pose
* Light Variations
* Variations of expression, age
* Partial occultation of the face (concealing)

These variations are the most difficult because the variations in the appearance of a person face according to different pose or light conditions are often far more important than the variation between face images of two different individuals acquired under the same conditions.

This explains why pictures should be taken in specific conditions so that facial recognition can be efficient.

* 1. **The methods used for face recognition**

Facial recognition methods can be classified into two broad categories: local and global methods. Amongst those methods, main ones will be presented thereafter.

* + 1. **Global methods**

Global methods are based on well known techniques of statistical analysis. In these methods, face images (which can be shown as matrices of pixel values) are used as input of the recognition algorithm and are generally transformed into vectors, which are easier to handle. The main advantage of global methods is that they are relatively quick to set up in. However, they are very sensitive to variations of illumination, pose and facial expression.

The main existing methods are:

The Principal Component Analysis (PCA) : **EigenFaces**

The LDA (Linear Discriminant Analysis) Algorithm : **FisherFaces**

* + 1. **Local methods (Geometric)**

The local methods include transformations applying to specific areas of the image, usually around characteristic points (corners of the eyes, mouth, nose, ...). Therefore, they require a priori knowledge on images. These methods are more difficult to implement but are more robust to the problems due to variations of illumination, pose and facial expression. The main existing methods are:

EBGM (Elastic Bunch Graph Matching);

Modular Eigenface;

Hidden Markov Method.

But in fact, our aim on this project will be obviously to use both main global methods.

Both methods that we will present are using a common training algorithm steps that are :

* Preprocessing of training image set
* Normalization \_mean image estimation
* Use of PCA/ LDA/ ICA

PCA/ LDA/ ICA are statistical tools used to implement facial recognition method. For instance the use of PCA is divided into two steps :

* The determination of the input image weight from projecting input image into the face space and by multiplying the resulted vector to eigenfaces of the database.
* A Comparison of results with metrics such as euclidian distance.

1. **EigenFaces**
2. Presentation of Eigenfaces

The Eigenface approach began with a search for a low-dimensional representation of face images. Sirovich and Kirby in 1987 showed that Principal Component Analysis could be used on a collection of face images to form a set of basis features. These basis images, known as Eigenpictures, could be linearly combined to reconstruct images in the original training set.The approach of using eigenfaces for recognition was developed by Sirovich and Kirby and used by Matthew Turk and Alex Pentland in face classification.

Eigenfaces is the name given to a set of eigenvectors when they are used in the computer vision problem of human face recognition. The eigenvectors are derived from the covariance matrix of the probability distribution over the high-dimensional vector space of face images. The eigenfaces themselves form a basis set of all images used to construct the covariance matrix. This produces dimension reduction by allowing the smaller set of basis images to represent the original training images. Classification can be achieved by comparing how faces are represented by the basis set.

1. Procedure

A set of eigenfaces can be generated by performing a mathematical process called Principal Component Analysis (PCA) on a large set of images depicting different human faces. Informally, eigenfaces can be considered as a set of "standardized face ingredients", derived from statistical analysis of many pictures of faces. Any human face can be considered to be a combination of these standard faces.

To create a set of eigenfaces, one must:

* Prepare a training set of face images. The pictures constituting the training set should have been taken under the same lighting conditions, and must be normalized to have the eyes and mouths aligned across all images. They must also be all resampled to a common pixel resolution (r × c). Each image is treated as one vector, simply by concatenating the rows of pixels in the original image, resulting in a single row with r × c elements. For this implementation, it is assumed that all images of the training set are stored in a single matrix T, where each column of the matrix is an image.
* Calculate the average image by adding each columns of the matrix T and dividing the previous obtained vector by the number of input images.
* Subtract the mean from matrix T to obtain matrix A (where each element represents the luminance variance of each pixel). Once the average image a is calculated and it is then subtracted from each original image in T.
* Calculate the covariance matrix S.
* Calculate the eigenvectors and eigenvalues of the covariance matrix S. Each eigenvector has the same dimensionality (number of components) as the original images, and thus can itself be seen as an image. The eigenvectors of this covariance matrix are therefore called eigenfaces. They are the directions in which the images differ from the mean image. Sort the eigenvalues in descending order and arrange eigenvectors accordingly.
* Choose the principal components. The number of principal components k is determined arbitrarily by setting a threshold ε on the total variance.

Total variance v = n\*(λ1+ λ2+…+ λn), n= number of data images

* k is the smallest number satisfies
* Determinate the input image weight determination from projecting each image.
* Each image is represented by a vector which is used to reconstruct the images. We then save the average image, eigenfaces and the projection (weight ) of images.

This ends the training part of the implementation, some skills are used

1. Benefits and deficiencies

* Benefits

Eigenface provides an easy and cheap way to realize face recognition in that:

* Its training process is completely automatic and easy to code.
* Eigenface adequately reduces statistical complexity in face image representation.
* Once eigenfaces of a database are calculated, face recognition can be achieved in real time.
* Eigenface can handle large databases.
* Deficiencies

However, the deficiencies of the eigenface method are also obvious:

* Very sensitive to lighting, scale and translation; requires a highly controlled environment.
* Eigenface has difficulty capturing expression changes.
* The most significant eigenfaces are mainly about illumination encoding and don't provide useful information regarding the actual face.

**Algortihm and steps**

n summary of the preceding description, the L-Fisherfaces algorithm is given below:

I

Step 1: Use PCA to transform the input space ℜN into an n-dimensional space ℜn , where n N. Since rank(Sw L ) ≤ M − L, where M is the number of training samples, and L is the number of classes. So, we choose n ≤ M − L to make Sw L nonsingular.

Step 2: Construct the within-class and between-class matrix: For the given training data projected into PCA subspace, construct the within-class matrix Hw and betweenclass matrix Hb using Eqs. (16) and (21), respectively.

Step 3: Construct the local within-class and between-class scatter matrices Sw L and Sb L in PCA subspace using Eqs. (19) and (24).

Step 4: Calculate the generalized eigenvectors α1, α2, …, αd (d ≤ n) of Sb L α = λSw L α corresponding to the d largest positive eigenvalues λ1 ≥ λ2 ≥ … ≥ λd. Let WPCA denote the transform matrix of PCA and A = (α1, α2, …, αd). WLFDE = WPCAA is the transformation matrix from the image space into the LFDE subspace. The column vectors of WLFDE are the so-called L-Fisherfaces. The feature vector extracted by LFDE is y = T WLFDE x = . T T A WPCA x Using the ORL face database as the training set, we present the first ten L-Fisherfaces in Fig. 1, together with Eigenfaces, Fisherfaces and Laplacianfaces [7]. It is interesting to note that the L-Fisherfaces are somehow similar to the combination of Fisherfaces and Laplacianfaces.

1. **FisherFaces**

Eigenface method uses PCA for dimensionality reduction, which yields projection directions that maximize the total scatter across all classes of images. This projection is best for reconstruction of images from a low dimensional basis. However, this method doesn’t make use of between-class scatter. The projection may not be optimal from discrimination for different classes.

While this is clearly a powerful way to represent data, it does not consider any classes and so a lot of discriminative information may be lost when throwing components away.

The Fisherface method is an enhancement of the Eigenface method that it uses Fisher’s Linear Discriminant Analysis (FLDA or LDA) for the dimensionality reduction. The LDA maximizes the ratio of between-class scatter to that of within-class scatter, therefore, it works better than PCA for purpose of discrimination. The Fisherface is especially useful when facial images have large variations in illumination and facial expression.

This projection maximizes the ratio of between-class scatter to that of within-class scatter. The idea is that it tries to “shape” the scatter in order to make it more reliable for classification.

* 1. **Linear Discriminant Analysis (LDA)**

The Linear Discriminant Analysis (LDA) is used to find the linear combination of characteristics that better separate object or event classes. The resulting combinations can be used as a linear classifier, or generally in reducing characteristics before the posterior classification.

LDA is closely related to the PCA, because both seek the linear combinations of the variables that better represents the data. This statistic skill explicitly to model the difference between data classes unlike the PCA which does not take into account the differences between classes.

* 1. **LDA for recognition**

The LDA by recognition algorithm is divided into two phases, one for the calculation of person models called system learning phase and the other which is to recognize a person thanks to registered models called test stage.

* + 1. **Learning phase**

As in the PCA, it gathers the images of the learning database in a large image matrix T where each column represents a image listed Ti, then the average image is calculated.

For each class C, the average image is calculated.

Each image Ti of each class Ci is then refocused in comparison of the average. This produces a new image.

Then we proceed at calculation of the different scatter (dispersion) matrixes named as followed:

* The Intra-class Distribution Matrix;
* The Inter-class Distribution Matrix;
* Total Dispersion Matrix.

After we have defined the different dispersion matrixes, we must find the best projection that maximizes the intra-class dispersion on its matrix while minimizing inter-class dispersion, also on its matrix.

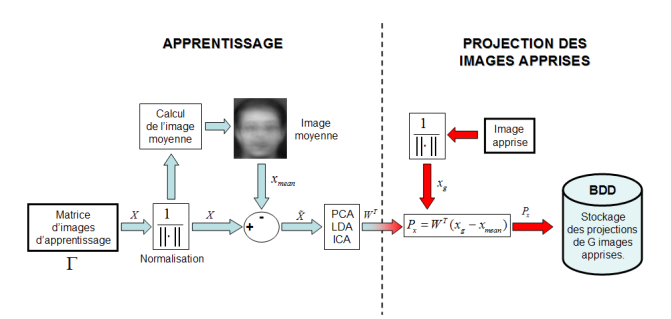


Fig : Learning phase of a face recognition system using a global method

* + 1. **Test phase or test stage**

Once the optimal projection that maximizes the found intraclass dispersion, the same pattern as the PCA on the projection of learned image and the projection of a test image is applied.

Then we project the test image in the Fisher space.

We compare the models obtained in learning phase. The comparison is made by calculating the distances (for instance, one can use calculation of the Euclidean Distance) between models and the test vectors and a decision rule is used to classify people.

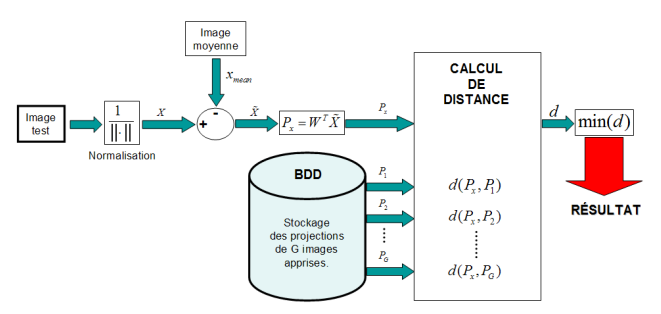


Fig : Test phase of a face recognition system using a global method

* 1. **Benefits and deficiencies**

The use of the LDA method for face recognition afford the following advantages:

* Maximizing inter-class scatter ;
* Reduction of intra-class scatter ;
* The method of Fisherfaces solves the problem of robustness to changes in pose, and facial expressions.

Despite these advantages, in the literature a serie of negative spikes still exists as:

* Costly (heavy) in computation time ;
* Costly (heavy) in memory space ;
* Makes poor results when the number of training images is great.

http://vision.ucsd.edu/kriegman-grp/papers/pami97.pdf

1. Project implementation
2. EigenFaces prototyping
3. FisherFaces prototyping
4. Guideline manual
5. Project management

CONCLUSION