Access control: adaptation and real time implantation of a face recognition method.

J. Mitéran, J.P. Zimmer, F. Yang, M. Paindavoine.

Laboratory Le2i, University of Burgundy, Aile des Sciences de l'Ingénieur, BP 400, 21011 Dijon, France

Email : miteranj@u-bourgogne.fr Tel : 33 (0)3 80 39 68 55 Fax : 33(0)3 80 39 68 54

Abstract

In this paper, an improvement of a face recognition method is proposed. The goal is to develop an easy to use access control software, allowing Personal Computer or building access control with minimal constraints for the users. This approach requires a high-speed classification method (about eight images/sec), and a high global recognition rate. We obtain good results using a method derived from Principal Component Analysis, a geometric transformation of the feature space, and a fast decision algorithm.

Keywords: Face recognition; Image processing; Access control

---

1. Introduction

To improve the security by controlling the access to PC computers or buildings (industrial, public or private), we adapted to public use and implemented in real time a robust and fast face recognition method. The aim of this paper is to present the adaptations needed for public use of this kind of control, and the performance evaluation of the modified method.

Numerous algorithms have been proposed for face recognition in the literature. Some of the most common methods are based on direct images of faces, known as Eigenfaces [1, 2, 3], on profiles [4], or on geometry [5, 6], or on 3D data acquisition of the surface of the face [7].

For access control of PC, it is necessary to have a low cost and fast system, requiring very few adjustments. The system must also be easy to use and with a minimum of constraints for the users. This excludes all active methods, based on the projection of a pattern, for example.

Regarding the performance in terms of recognition and computational time, we have first chosen in our laboratory to implement the method based on the Eigenfaces, which uses principal component analysis (PCA) for dimensionality reduction. For the classification and recognition step, we based our implementation on a neural network. A fast implementation of this method is described in detail by Yang et al. [9,16], and then J.P. Zimmer et al. [10]. The system is composed of a standard video conferencing camera, connected to a PC via either the parallel port, or a PCI acquisition board. We present in this paper an adaptation of this method to the systems, where the computational time is critical, and where the user can enter his name or a code before proceeding to the recognition test. That means that we can just perform a verification of the identity. In terms of performance, improvement was necessary for our applications (security access), where a very low rate of confusion is needed, but where the success identification rate is not so critical.

In the first part of this article, we recall the theoretical basis of the Eigenfaces method. In the second part, we present the problem caused by the learning and decision protocol, and the solution we have chosen. In the last part, we present the results of the implementation.

2. Recall of the eigenvector based method used for face recognition

This general face recognition method, implemented by Yang[16], is based on the PCA and perceptron. It is composed of two phases: a learning phase during wich approximately ten images of each user must be stored in the database and the decision phase, during which a new image must be acquired and classified as unknown or as a match for an existing user.

2.1. Learning Phase

During the first step, we create the face image database used to compute the input stimuli of the learning network. We based our choice on the method described by Abdi [11, 12], and we decompose each image (width L and height l), into a singular values vector.

The advantage of this method is to present small-sized vectors to the first layer of the network. For that, initially, we build a human base matrix $\mathbf{H}_{[l,L,n]}$, in which *n.p* face images are organized in a row order. **H** is constructed with all the images of *n* human faces to be recognized (*p* images per person).

This matrix can be decomposed in singular values as follows:

$$\mathbf{H} = \mathbf{P}\Lambda\mathbf{Q}^T \tag{1}$$

with:

Δ: Diagonal matrix of singular values $= \Lambda^{1/2}$,

Λ: Matrix of eigen values of $\mathbf{HH}^T$ and $\mathbf{H}^T\mathbf{H}$,

$\mathbf{P}$: Matrix of eigenvectors of $\mathbf{HH}^T$,

$\mathbf{Q}$: Matrix of eigenvectors of $\mathbf{H}^T\mathbf{H}$.

Determining **P** directly by computing $\mathbf{HH}^T$, of dimension [*l.L, l.L*] requires an inordinate amount of memory. Therefore, we first determine the matrix $\mathbf{Q}$. Secondly, we compute $\mathbf{P}$ with:

$$\mathbf{P}=\mathbf{HQ}\Delta^{-1} \tag{2}$$

Each stimulus $\mathbf{x}_k$ presented to the network (Fig. 1) is the result of the projection of the k^{th} face image vector onto the $\mathbf{P}$ matrix (k=1, …, $n.p+u$, where $n.p$ is the image number of **H**, and u the image number of "unknown" persons).

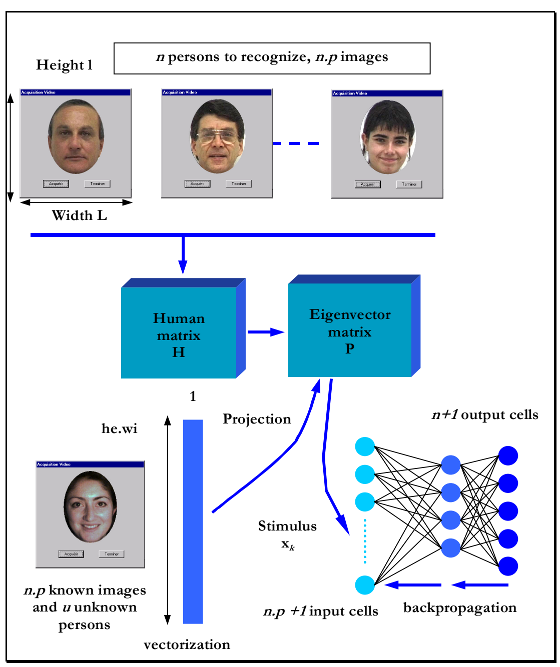


Fig. 1 Learning phase in case of recognition

The second step is the determination of the neural network parameters. The learning structure we use is a network with one hidden layer as in Solheim et al.[13] and Frasconi et al.[14]. Therefore, for the k^{th} stimulus $\mathbf{x}_k$, we define, $\mathbf{h}_k$ as the output value of the hidden layer, $\mathbf{o}_k$ as the value of the output layer of the network, and $\mathbf{r}_k$ as the theoretical value of the output layer of the network, assuming that the membership of $\mathbf{x}_k$ to a given class is known.

If $\mathbf{W}$ is the matrix of the weight connections from the input layer to the hidden layer, and $\mathbf{Z}$ the matrix of weight connections from the hidden layer to the output layer, the cells of the hidden layer compute their activation and convert it into a response by using their transfer function, according to:

$$\mathbf{h}_k = f\,(\mathbf{W}\mathbf{x}_k) \tag{3}$$

Then, activation of the output layer is computed using $\mathbf{Z}$ and the transfer function f included in each cell according to:

$$\mathbf{o}_k = f\,(\mathbf{Z}\mathbf{h}_k) \tag{4}$$

f is a non-linear differentiable function. We use the common logistic function[12].

We have chosen the known backpropagation algorithm in order to compute the learning parameters. This method is used to modify in a iterative way, the weights of the connections $\mathbf{W}$ and $\mathbf{Z}$ in order to reduce the final error value of classification. The error $\mathbf{e}_k$ obtained at the output layer is estimated by comparing the results given by the $\mathbf{o}_k$ cells with the theoretical response $\mathbf{r}_k$ of this layer:

$$\mathbf{e}_{k} = \mathbf{r}_k - \mathbf{o}_k \tag{5}$$

The output error signal $\boldsymbol{\delta}_{s,k}$ taking into account the error due to the cell and of its activation state is given by:

$$\boldsymbol{\delta}_{s,k} = f'\,(\mathbf{Z}\mathbf{h}_k) * \mathbf{e}_k. \tag{6}$$

Where $*$ is Hadamar's product.

The connection matrix $\mathbf{Z}$ is corrected at each iteration and becomes for the t^{th} iteration:

$$\mathbf{Z}_{t+1} = \mathbf{Z}_t + \eta\, \boldsymbol{\delta}_{s,k}\, \mathbf{h}_k^{\mathrm{T}} \tag{7}$$

The error propagates then on the hidden layer, and we estimate the error signal $\boldsymbol{\delta}_{h,k}$ for the cells of this layer by:

$$\boldsymbol{\delta}_{h,k} = f'\,(\mathbf{W}\mathbf{x}_k) * (\mathbf{Z}_t^{\mathrm{T}}\boldsymbol{\delta}_{s,k}) \tag{8}$$

The connection matrix $\mathbf{W}$ is corrected for the iteration $t+1$ according to:

$$\mathbf{W}_{t+1} = \mathbf{W}_t + \eta\, \boldsymbol{\delta}_{h,k}\, \mathbf{X}_k^{\mathrm{T}} \tag{9}$$

Then the algorithm iterates until we get the desired minimum error (a threshold is fixed by the user).

2.2. Decision Phase

During this step, an image of the face is acquired. Then, the image is vectored and projected onto the eigenvector matrix $\mathbf{P}$. The projection result is used as a stimulus to the network, which gives a response (set of $n+1$ values of the output cells). Finally, a decision threshold is used in order to eliminate the weakest values.

2.3. Performance of the method

Performance of the Eigenfaces method is widely described in the literature (Ref. 15, for example). In order to estimate our own implementation, including the acquisition system and the learning protocol, we studied the performance using a database of 250 images [16]. This database was divided in subsets D1, D2 and D3:

- D1 contains the images of 10 persons to recognize (10 images per person) used to compute the Eigenvectors (Fig. 2).

Fig. 2 : Extract of D1

- D2 contains the same images as D1, and 5 "unknown" persons (2 images per person). These images allow to compute the neural network parameters (Fig. 3).

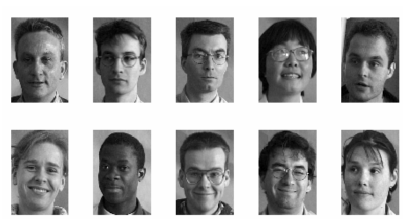


Fig. 3 : Extract of D2

- D3 contains 140 images, with 10 images per known person (10 persons), 4 images per "unknown" person, and 4 images per new person (5 persons), never learned by the system (Fig. 4).

Fig. 4 : Extract of D3

The best global error rate we obtained concerning confusion (known or unknown persons classified as other persons) is 2%, but if we take account the proportion of unknown persons in the D3 base, 10 % of unknown persons are classified as known persons. These performances were not acceptable in the case of access control. We must then adapt the method.

3. Adaptation to face verification

3.1. Data organization

In the case of access control on a PC, the user must enter his name and password before access is granted. Since the system knows the name of the individual, we just need to verify his identity, not to distinguish him from all of the other users. Therefore, we can separate all of the persons to be identified, using one set of data (**H**, **P**, **W**, **Z**) per person (Fig. 5).

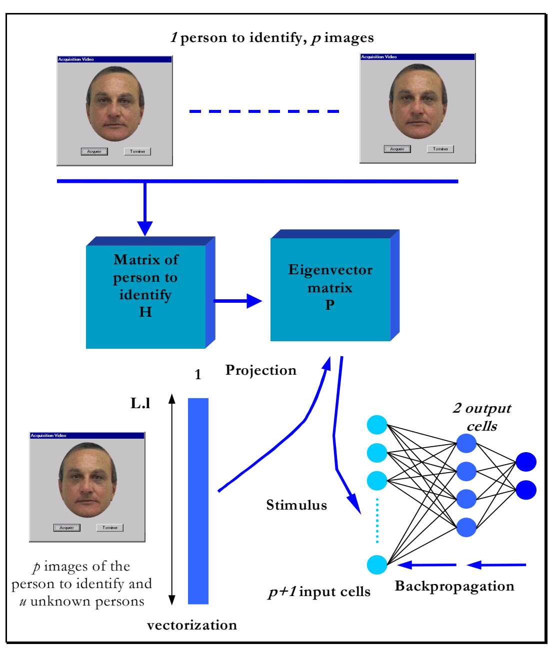


Fig. 5 Learning phase for one person in case of verification

This allows the decision phase to be very fast. In the case of access control of buildings, the person must hit a button corresponding to his name. This choice allows an unlimited number of persons to be recognized. The practical limit is the disk space used to store the learning images and matrices **H**, **P**, **W**, **Z** (about one megabyte per person).

3.2. Definition of the acquisition protocol

In the first version of the face recognition software, the decision protocol was defined as follows. The user, during the recognition phase, must hit a key of the keyboard to grab his face after positioning. The person was accepted or not using only this one image. This protocol required a high success rate during the recognition phase. Moreover, it was not practical to use.

We chose the following protocol:

1- The user enters his name.

2- The set of data corresponding to the name is loaded in memory, and the test number T is set to 0.

3- One image is acquired, and the verification test is performed. T is incremented.

4- If the test succeeds, the user is authorized and the protocol stop. If the test failed, the process loops to the step 3, while T<100.

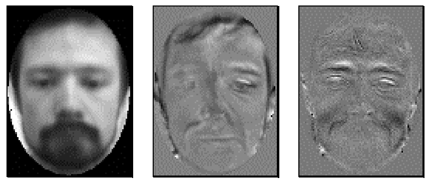
5- If the user is still non-recognized after T=100 tests, the protocol stops and the user is rejected.

This is possible because approximately eight images per second, in terms of recognition, can be handled by the system (these results were obtained using a Pentium II 233 MHz processor, with 64MB of memory, and a standard PCI acquisition board based on the BT848 chip). This allows the system to discard misaligned images but also increases the probability of encountering a classification error: a *confusion*, which occurs when an unknown user is identified as a known user..

We modified the method regarding these new features. We reduced the confusion rate, which also increased the per-image non-detection rate. However, the large number of images processed rendered this increase in the non-detection rate insignificant.

3.2.1. Diminution of confusion rate

For a better understanding of the problems that the system must overcome, it is necessary to recall that the first eigenvector represents the low frequencies in the image, and the last eigenvectors are representative of high frequencies. These high frequencies contain the most significant features of an individual (Fig. 6).



First eigenvector – fifth eigenvector – tenth eigenvector

Fig. 6. Eigenvector representation

Here, the projections are computed using the luminance of the images, causing two problems. Firstly, image acquisition quality depends on external lighting conditions, which is not crucial because the camera has good automatic gain control. Secondly, if someone, in the database has a particular feature (beard, for example), where the luminance takes a dominant place, the neural network will converge regarding mainly this first eigenvector, and thus regarding low frequencies only. This implies that his features are very different from those features of the unknown persons in the learning data set, and if the system attempts to recognize another bearded person, there will be some confusion (Fig. 7).

To solve this problem, we studied two methods. The first method, inspired from Belhumeur et al.[15], consisted in the elimination the three first eigenvectors to perform the classification. The second method, inspired from Yang[16], was to use edge detection (low pass filter followed by a Sobel filter) instead of using direct luminance (Fig. 8).

Fig. 7. Example of confusion

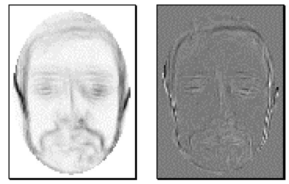


Fig. 8. First and tenth eigenvectors after edge detection

The two methods gave some similar results regarding confusions, since they eliminate all confusions in our test. For practical reasons, we choose to implement the second method, which avoids lost of information that may be useful for discrimination.

This choice allows the method to be globally more robust (the rate of false alarms is reduced), even if the recognition rate per individual image is less than if using direct luminance. We do not give here any rate of success, because the system must be evaluated globally, including the dynamic aspect of the acquisition and classification (see the results given in the last part).

3.3. Problems of convergence

Since we eliminate low frequencies, the method is now more discriminatory using the high frequencies of the images, which represent the identity of the person. However, the main problem of neural networks, well known in the literature, is that sometimes, depending on the person to be recognized, the backpropagation algorithm does not converge. The main reason is that the high-level eigenvectors distributions are not easily separable in the feature space. To solve this problem, it is known that we can improve the neural network, choosing more hidden layer, for example, but this solution was not chosen, mainly for computing time reasons. Moreover, since the distribution has a particular symmetry in the feature space, it is simple to compute a transformation using this symmetry. We have represented in Fig. 9 an example of projections of the learning images on the 5^{th} and 6^{th} eigenvectors (high level eigenvectors).

The known user is represented by squares, and unknown persons by triangles. One can see that the samples corresponding to the known user are randomly distributed around the samples corresponding to unknown persons.

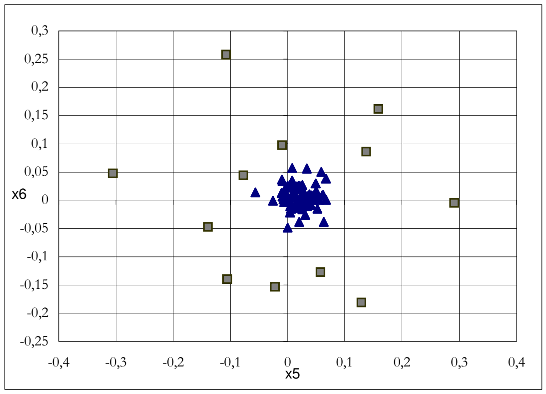


Fig. 9. Projection of 5^{th} and 6^{th} eigenvectors

After investigating the geometrical shape of the distributions, we choose to keep the first three projections without any modification, and the fourth feature of the classification method is computed as:

$$\mathbf{x'}_3 = \sqrt{\sum_{i=1}^{n.p} x_i^2}$$

Where $\mathbf{x}_k$ is the projection of the image on the k^{th} eigenvector. This method allows to obtain the same results as a circular threshold around the distribution of unknown persons, but is better suited to the classification method.

We represent on Fig. 10 and Fig. 11 an example of the distribution obtained with one person to be recognized.

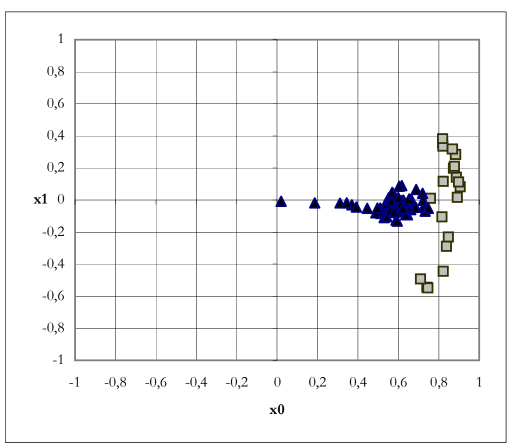


Fig. 10. Projection on 1^{st} and 2^{nd} eigenvectors

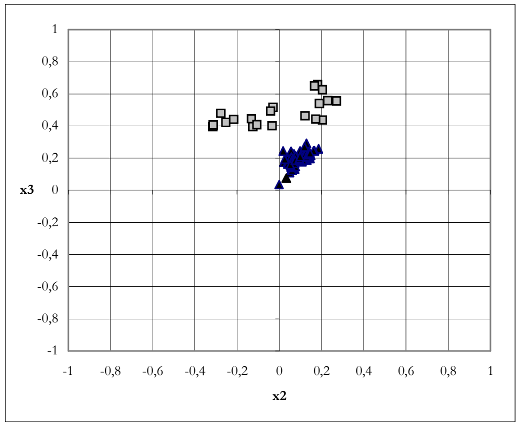


Fig. 11. Projection on 3rd eigenvector and x'3

The figures shows that the two classes are now very easily separable. Moreover, the decision phase is now faster, since the number of cells in the first layer of the neural network no longer depends on the number of learning images of one person.

4. Results of the improved method

Finally, we have tested our software in the case of face verification. The first test was made with a fixed lighting. The second test was performed to illustrate robustness against lighting variations.

4.1. Standard test

In order to evaluate the performance of the system, we defined an acquisition protocol, which reflects the real use of the software. All the measures were made in real conditions: the user acquires himself the images of the learning phase, and performs the test himself by presenting his face in front of the camera.

For practical reasons, we divided a set of 30 persons to be recognized in 6 subsets of 5 persons each. The steps of the learning and recognition protocol are as follows:

1 - Record 5 images of each person in the subset.

2 - Record 5 new images of each person. The separation between the first and the second step allows to take into account positional variations.

3 - Execute the learning phase (computation of **H**, **P**, **W**, **Z** for each person).

4 - Perform 3 tests for each person (verification during 100 images), so that he knows how to present himself in front of the camera in order to be well recognized. Add a few images if necessary. The maximum number of learning images is fixed to 15.

5 - Perform for each person alternatively 5 identification tests entering his real name, and 5 tests of intrusion using a false name.

The results of the test are reported in Fig. 12. One can see that we obtain a 90% success rate in terms of verification. It is important to note that 100% of users were well identified at least one time, and that the success rate is better for experienced users (96%). The most important result is that we obtained only two confusions. After analysis, the origin of these confusions was in the bad positioning of the person during the learning phase. One can note that the quality of the learning phase, as in all classification problems, determines the performance of the system. If someone moves in front of the camera during the acquisition phase, the registered image will contain only a part of his face, and the risk of intrusion for this individual increases. The user must verify each learning image before accepting or rejecting it. After removing the bad images, the two confusions of the test disappeared. This fact is usually not taken into account in the literature, where authors often test their face recognition using a fixed database of picture. The originality of this protocol is to characterize all the system, including face positioning during image acquisition and final decision.

Number of verifications	150
Failed	15
Success	135
Number of confusion tests	150
Success (person not authorised)	148
Failed : (person authorised)	2

Fig. 12 Results of the test

4.2. Lighting variation test

To illustrate the robustness of the method, we performed the test using two different learning set.

The first learning set is built using only one standard homogenous lighting (Fig. 13).

Fig. 13 Learning with homogenous lighting

The second learning set is built using four different lighting: homogenous, left illumination, right illumination, and globally darken (Fig. 14). We have chosen to test only realistic variations one can find in an office, for example.

Fig. 14 Learning with lighting variations

For both learning sets, we tested the identification with homogenous lighting, and with lighting variations. To illustrate the improvement of performance using edge detection instead of direct luminance, we also tested the identification using both algorithms.

Result using direct luminance		**Example of images**
Learning with homogenous lighting	Success rate **23%**	
	Error rate **77%**	
Learning with lighting variations	Success rate **53%**	
	Error rate **47%**	

Fig. 15 Results using luminance

The results (performed with five experimented users and five tests of each lighting for each user) are reported in Fig. 15 and Fig. 16.

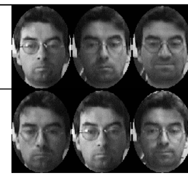
Result using edge detection		**Example of images**
Learning with homogenous lighting	Success rate **56%**	
	Error rate **44%**	
Learning with lighting variations	Success rate **96%**	
	Error rate **4%**	

Fig. 16 Results using edge detection

One can see that the method using edge detection is more robust than using direct luminance, but it is necessary to take into account important lighting variations during learning phase to obtain the best results.

We performed also some qualitative tests showing that the result is not influenced by the glasses, and little influenced by hair style if the image registration is made carefully[16] (the hair is not taken into account thanks to the elliptic mask). To avoid a too great sensitivity to the slow variations of beard, we added the possibility of introducing automatically the image having allowed the last recognition into the learning base. This one thus is automatically updated. However, the method remains sensitive to abrupt variations of beard.

5. Conclusions

We have implemented a face verification software, used for PC or building access control, improving the feature computation and separability of the learning sets in the feature space.

In terms of performance, we showed that each person can be easily recognized and that the software can be used in a real environment.

We have also shown that the measurement of the recognition rate and the adjustment of the classification method must be adapted to the acquisition protocol to be significant.

Our future goal is to eliminate the positional dependence of the user in front of the camera, implementing an automatic detection and adjustment of the face position. We are currently developing a mixed method, based on motion detection and the Hough[17, 18] transform.

Finally, the improved method can be used in several areas, protecting PC or building access, combining statistical and neural approaches.

Acknowledgements

Thanks to H. Abdi for his advice on neural networks, and to Alan Mangan and Lew Lew Yan Voon for their help with the English translation.

References

1. L. Sirovitch, M. Kirby, "Low-dimensional procedure for the caracterization of human faces", J. Optical Soc. of America, **2**, 519-526 (1987).
2. M. Turk, A. Pentland, "Eigenfaces for recognition", J. of Cognitive Neuroscience, **3**(1), 71-86 (1991).
3. A. Pentland, B. Moghaddam, O. SDtarner, T. Oliyide, M. Turk,."View based and modular eigenspaces for face recognition", Computer vision and pattern recognition, 84-91 (1994).
4. L. Najman, R. Vaillant, E. Pernot,. "Face Sideviews to identification", Ginanni Vemazza editor, Images Processing,: theory and applications, San Remo, Italy (1993).
5. R. Brunelli, T. Poggio,. "Face recognition : Features versus template", IEEE Trans. on PAMI, 1042-1052 (1992).
6. A. L. Yuille, D. S. Cohen, P. W. Hallinan,. "Feature extraction from face using deformable templates", IJCV, **8**(2), 99-111 (1992).
7. J. Y. Cartoux, *Formes dans les images de profondeur. Application à la reconnaissance et à l'authentification de visages*, PH thesis, Université Blaise Pascal de Clermont-Ferrand (1989).
8. R. Vaillant, I. Surin. "Face reconstruction through active stereovision", Traitement du signal, **12**(2), 201-210, (1995).
9. F. Yang, E. Drege, M. Paindavoine, H. Abdi,."Parallel Implementation of a Face Location Algorithm Based on the Hough Transformation", EUSIPCO-Greece, 495-498 (1998).
10. J. P. Zimmer, J. Mitéran, F. Yang, M. Paindavoine, "Security software using neural networks", IECON 98, Aachen, Germany, 72-74 (1998).
11. D. Valentin, H. Abdi, A. J. O'Toole, "Categorization and identification of human face images by neural networks: A review of the linear autoassociative and principal component approaches" , Journal of Biological Systems*, 2*, 413-429 (1994).
12. H. Abdi, *Les Réseaux de neurones*, Presses Universitaires de Grenoble, France (1994).

13. I. Solheim, T. L. Payne, R. Castain,. "The potential in using backpropagation neural networks for facial verification", Simulation, **58**(5), 306-310 (1992).
14. P. Frasconi, M. Gori, et al,. "Face recognition using Multi-Layered Perceptron which also reject never seen faces", Proceedings of 4th Italian workshop on Neural Networks and Parallel Architectures, Vietry, Italy, 58-66 (1992).
15. P. Belhumeur, J. Hespanha, D. Kriegman, "Eigenfaces vs. Fisherfaces : Recognition using class specific linear projection", IEEE, trans. on **PAMI**, **19**(7), July 1997, 711-720 (1997).
16. F. Yang, "Traitement automatique d'images de visages. Algorithme et architecture.", Thèse, Université de Bourgogne, (1998).
17. P.V.C. Hough, *Methods and means for recognising complex patterns*, US patent 3069654, (1962).
18. Y. Lei, K. C. Wong, "Ellipse detection based on symmetry", Pattern recognition letters, 20, 41-47. (1999).

Figure captions

Fig. 1 Learning phase in case of recognition
Fig. 2 : Extract of D1
Fig. 3 : Extract of D2
Fig. 4 : Extract of D3
Fig. 5 Learning phase for one person in case of verification
Fig. 6. Eigenvector representation
Fig. 7. Example of confusion
Fig. 8. First and tenth eigenvectors after edge detection
Fig. 9. Projection of 5th and 6th eigenvectors
Fig. 10. Projection on 1st and 2nd eigenvectors
Fig. 11. Projection on 3rd eigenvector and x'3
Fig. 12 Results of the test
Fig. 13 Learning with homogenous lighting
Fig. 14 Learning with lighting variations
Fig. 15 Results using luminance
Fig. 16 Results using edge detection