Array of Multilayer Perceptrons with No-class Resampling Training for Face Recognition

D. Capello[1], C. Martínez[2,3], D. Milone[2] and G. Stegmayer[1]

[1] CIDISI-UTN-FRSF, CONICET, Lavaise 610 - Santa Fe (Argentina)
[2] sinc(*i*)-FICH-UNL, CONICET, Ciudad Universitaria UNL - Santa Fe (Argentina)
[3] Laboratorio de Cibernética-FI-UNER, C.C. 47 Suc. 3-3100, Entre Ríos (Argentina)
dcapello@santafe-conicet.gov.ar

Abstract. A face recognition (FR) problem involves the face detection, representation and classification steps. Once a face is located in an image, it has to be represented through a feature extraction process, for later performing a proper face classification task. The most widely used approach for feature extraction is the eigenfaces method, where an eigenspace is established from the image training samples using principal components analysis. In the classification phase, an input face is projected to the obtained eigenspace and classified by an appropriate classifier. Neural network classifiers based on multilayer perceptron models have proven to be well suited to this task. This paper presents an array of multilayer perceptron neural networks trained with a novel no-class resampling strategy with takes into account the balance problem between class and no-class examples and increase the generalization capabilities. The proposed model is compared against a classical multilayer perceptron classifier for face recognition over the AT&T database of faces. The results obtained show interesting results regarding the improvement in classification rates.

1 Introduction

Over the last years, face recognition (FR) has become one of the most popular biometric technologies. Its objective is to automatically identify a person from a picture or a frame in a video source [1]. A complete automatic FR system includes the following stages: face detection, face representation and face classification. Face detection refers to finding a human face in the scene by means of image processing techniques. The face is then represented through an appropiate feature extraction method, which extracts useful information for the classification. The final stage corresponds to the recognition itself using some classifier, designed according to the previous extracted data.

The feature extraction methods can be divided into two main classes: a global representation by means of an holistic face encoding and a local component representation using the geometry of facial features [1]. In the first group, a widely used technique is the eigenfaces [2], which consists of the application of the principal component analysis (PCA) [3] for dimensionality reduction. This technique obtains the feature vectors with a mapping of each image into a space previously established by PCA. The methods of the second group generally involve more complex techniques to locate and measure facial components, finding geometric relationships that add robustness against pose changes [4].

The classification is then carried out using an appropiate classifier. For this task, different models have been proposed, from simple statistical pattern recognition approaches like Euclidean distance or k-nearest-neighbors up to ensemble solutions based on hundred of weak classifiers [5]. Neural networks have also proved to be well-suited for this problem, with the multilayer perceptron (MLP) being one of the most popular models [6].

In this work, an array of neural networks is proposed along with a novel training algorithm to approach the class and no-class examples balance problem and the overfitting. The proposed classifier consists of one MLP for each subject (valid or authorized person), with a final decision made over the network outputs of the complete array. To overcome the no-class balance problem and increase the generalization capabilities, a resampling training procedure is proposed and applied to each MLP during training. Resampling [7] is any of a variety of methods for using subsets of available data, randomly selecting training examples from a set of data points. Our proposal is a special case, performing it only over no-class examples instead of all training patterns, and without changing their probability of being chosen. The novel algorithm here presented involves the use of different training patterns at every epoch in order to present all negative samples to the MLP. The model and training algorithm has been compared against a classical MLP classifier, significantly improving the classification rates in the comparison of several possible model configurations.

The outline of the paper is as follows. Section 2 reviews the previous work on face recognition using neural networks, mainly the investigations closer to the approach here presented. In Section 3, the formulation of the eigenfaces as a feature extraction method is outlined. Section 4 describes the proposed model: the fundamentals of the architecture and operation of the array of neural networks, together with the novel no-class resampling algorithm. Section 5 presents the face database used and the designed experiments, together with a discussion of the results obtained. Finally, Section 6 presents the conclussions and outlines future research.

2 Related work

Among appearance-based methods, the PCA-based linear projection of the complete face into a subspace, also named aigenfaces method [2], is the most widely applied approach to the feature extraction problem. As an alternative, independent component analysis (ICA) minimizes high-order statistical dependency on the input data and performs better than PCA on changes in expression, face across days or partial oclussions [8, 9], but PCA outperforms ICA when the data set is small [10]. The linear discriminant analysis (LDA) [11] also obtains a linear projection from the input data but maximizing the discrimination between classes. A generalization of the previous linear methods was proposed with the use of kernel methods [12, 13] and combined approaches [14]. An extension of PCA to 2D was presented in [15], which obtains a covariance matrix much smaller than the 1D case.

From these global features, neural networks have been proposed for classification in several studies. The MLP feedforward neural network can be trained with the backpropagation algorithm to classify the eigenfaces [16, 6]. It has been also applied to combinations of global features like PCA and LDA in a recent work [17]. In order to get better performance in difficult situations, like illumination changes and occlusions, novel investigations use the MLP to classify the eigenfaces obtained from infrared thermal images [18]. Among many other neural network models, the radial basis function (RBF) is a model extensively used, mainly with local features as input patterns [19, 20].

Classifiers based on multiple neural networks –the *modular networks*– were also applied to face recognition. In such systems, the class label is assigned with a final decision calculated from the outputs of each single classifier. In this context, modular networks based on MLP have been proposed for different tasks, that is, locate a face in the image [21] or recognize a face from an unseen view [22].

New trends in the application of neural networks to biometry include multimodal systems that combine face recognition with other biometric identification techniques, such as speech recognition [23], or the more recent approach based on 3D imaging [24].

3 Feature extraction using principal component analysis

Principal component analysis or Karhunen-Love transformation is a standard technique used in statistical pattern recognition for dimensionality reduction. It applies an orthogonal linear transformation that decorrelates variables, retaining the ones that most contribute to the maximum variance contained in the pattern –the principal components– and discarding the rest [25].

A grayscale image with size W in width and H in height can also be considered as one dimensional vector of dimension $N = W \times H$. Considering a group of images with the same configuration, each of them corresponds to a point in a N-dimensional space. In this way, if we deal with aproximately similar images of faces, they will be located in a small region of the space. Here, the aim of PCA is to select a lower dimensional subspace that best represents the original images. As we will see, the new vectors are given by the eigenvectors of the covariance matrix corresponding to the original images, and because of its appearance they are refered to as *eigenfaces*.

In order to calculate the eigenfaces, first of all we need a training set of M face images $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2, \ldots, \mathbf{\Gamma}_M$, ideally of a well-known set of subjects in the database. Each $\mathbf{\Gamma}_i$ is a N-dimensional vector containing the N pixels of the i-th image.

The goal is to calculate the eigenvectors $\mathbf{u}_i$ of the covariance matrix of the $\mathbf{\Gamma}_i$. Defining the mean face as $\mathbf{\Psi} = \frac{1}{M}\sum_{i=1}^{M}\mathbf{\Gamma}_i$ and the difference between each image and the mean face as $\mathbf{\Phi}_i = \mathbf{\Gamma}_i - \mathbf{\Psi}$, the covariance matrix is given by

$$C = \frac{1}{M}\sum_{i=1}^{M}\mathbf{\Phi}_i\mathbf{\Phi}_i^T, \tag{1}$$

resulting a large $N \times N$ matrix even for small image sizes.

Calculating the eigenvectors of (1) is a computational intensive task. The method described in [2] is commonly used, which defines $A = [\mathbf{\Phi}_1\mathbf{\Phi}_2 \cdots \mathbf{\Phi}_M]$ as a $N \times M$ matrix (N rows, one for each image pixel, and M columns, one for each subject). This way, its covariance matrix AA^T has a size $N \times N$ but A^TA is a $M \times M$ matrix. Then, defining

$$L = A^TA \tag{2}$$

where

$$L_{ij} = \mathbf{\Phi}_i^T\mathbf{\Phi}_j \tag{3}$$

and calculating the M eigenvectors $\mathbf{v}_i$ of L, the eigenfaces are calculated as

$$\mathbf{u}_i = \sum_{j=1}^{M}\mathbf{v}_{ij}\mathbf{\Phi}_j, \qquad i = 1, \ldots, M. \tag{4}$$

Also, $\mathbf{u}_i$ eigenvectors can be ranked by their associated eigenvalue, and the first M' higher eigenfaces can be chosen according to the desirable proportion of total image variance to be represented. Generally, the number of training images will be smaller than the number of pixels in the images ($M \ll N$), so the number of operations are significantly reduced.

Any input image $\mathbf{\Gamma}$ can be projected in a M'-dimensional space (the "face space") by

$$\mathbf{\Omega} = \mathbf{u}^T(\mathbf{\Gamma} - \mathbf{\Psi}). \tag{5}$$

Hence, $\mathbf{\Omega}$ forms the $M'-$dimensional feature vector that represents the image $\mathbf{\Gamma}$ and that can be used as input in a classifier with fixed input size regardless of the number of pixels N of the original picture.

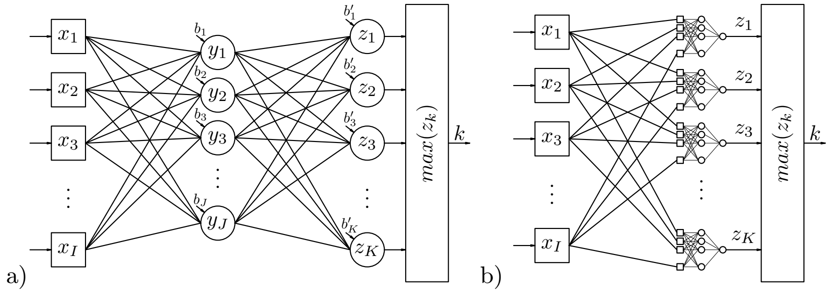

Fig. 1. a) Classical MLP classsifer. b) Proposed array of MLPs (aMLPs) model.

4 Proposed model architecture and training

4.1 Array of Multilayer perceptrons classifier

In this work we propose a classifier based on an array of feedforward multilayer perceptron networks. The model is an array of MLPs (aMLPs), where there is one MLP model for each subject k to be recognized, with $k = 1 \ldots K$ being K the total number of subjects. Therefore, the aMLPs model is formed by K networks like the one shown in Figure 1.b). Each network output should take a value of 1 if the subject is recognized or 0 otherwise. This array of classifiers has been compared against a classical MLP having K output neurons, one for each subject to be recognized (see Figure 1.a).

The first layer of each MLP in the aMLP is a set of I input neurons, where each input is a eigenspace point ($\mathbf{\Omega}$) such that $I = M'$, and there are J hidden neurons. When an image $\mathbf{\Gamma}$ has to be classified, its projected eigenspace point $\mathbf{\Omega}$ is used as input for the classifier, that is to say, it is presented to all the K networks defined for the aMLPs model, and the maximum output obtained among all network outputs is assigned a class label. If a picture of the k^{th}-subject has been presented to the model, a value of (near) 1 is expected at the k^{th} network.

The model parameters (weights and biases) are randomly initialized between -1 and 1 at the beginning of the training phase. Each neuron in the hidden and output layer has a sigmoid activation function. All the MLPs are trained with the standard backpropagation with momentum algorithm [26].

4.2 No-class Resampling Training Strategy

Two training strategies have been applied. The first one is the basic approach and involves training each classifier with 8 images of the class subject, plus 8 images of each of the remaining subjects (39). The second procedure is a novel

Algorithm 1: No-class resampling training procedure for the aMLPs: fixed epochs stopping.

```
Data:
    $K$: number of MLPs in the array (also number of classes)
    $N$: number of positive training samples
    $R$: negative/positive training samples ratio
Results:
    $\boldsymbol{\Theta}$: trained array of $K$ MLPs
begin
    for $1 \leq k \leq K$ do
        clear the training set $T$
        add $N$ positive samples of class $k$ to $T$
        for $1 \leq i \leq 10$ do
            initialize the network at random
            repeat
                pick a different set of $N \times R$ no-class samples and add them in $T$
                train 10 epochs with shuffling over $T$
                remove the no-class samples from $T$
            until all negative classes covered
        $\Theta_k \leftarrow$ network with minimum MSE
end
```

approach that tries to improve model generalization resampling over the negative examples for each class at each training epoch, and at the same time overcoming the class and no-class imbalance problem

The proposed no-class resampling training (NCRT) involves changing, dynamically on training time, the no-classes examples for each classifier. Each network is trained with 8 positive training patterns for the subject (class) and $8 \times R$ negative patterns, formed by 8 pictures of R other subjects (no-class). After 10 epochs, the R subjects are changed until the whole set of pictures has been sampled. A summary of the implemented method can be found in the pseudo-code of Algorithm 1. A variant of this algorithm has been also implemented, which simply involves iterating the training algorithm until a predefined mean square error (MSE) is reached, instead of resampling the no-class examples space after a fixed number of epochs.

5 Results and discussion

This section presents the data used for testing the proposed aMLPs model, the experimental design, the obtained results and their discusssion.

Fig. 2. Samples face images from the AT&T database of faces.

5.1 Face image database and feature extraction

For our experiments, the AT&T Laboratories database of faces (formerly, also referred to as 'The ORL database of faces') has been used[1]. It has been selected because it is widely used in the face recognition literature [27].

In this database, there are 10 different images of each of 40 persons of different gender, ethnic background and age (see Fig. 2). For some subjects, the images had been taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses/no glasses). All the images had been taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). The complete database contains 400 grayscale face images of size 92×112 pixels.

When the eigenfaces are ranked by magnitude of their corresponding eigenvalues, only the top M' eigenfaces need to be used. In [28] a limited version of the AT&T database of faces ($M = 115$) was evaluated and 40 eigenfaces were sufficient for a very good description of the face images. However, since we are using the whole set of subjects and pictures of the database ($M = 400$), the number of sufficient eigenfaces has been determined experimentally. We have used the method explained in Section 3 for choosing a subset of M' eigenfaces between $M = [1, 400]$, according to three levels of variance among the images. Several experiments were performed for representing, approximately, the 75%, 80%, and 85% of variance of all the images database, which correspond to $M' = 25$, $M' = 50$, and $M' = 75$ eigenfaces (inputs to the aMLPs model), respectively.

5.2 Experimental Results

The proposed aMLPs, with $K = 40$ networks, versus a classical MLP model with K output neurons have been tested using as comparative measurement

[1] AT&T Laboratories Cambridge, The ORL Database of Faces. http://www.cl.cam.ac.uk/research/ dtg/attarchive/facedatabase.html

Table 1. Comparison between a classical MLP classifier and the proposed aMLPs model. NCRT strategy for a fixed number of epochs: 10 epochs, $R = 3$.

Neurons	MLP+basic	aMLPs+basic	aMLPs+NCRT
I=25, H=25	66.55%	72.85%	72.20%
I=25, H=50	69.80%	72.37%	71.77%
I=25, H=75	68.75%	72.10%	70.85%
I=50, H=25	84.35%	88.42%	89.05%
I=50, H=50	85.25%	88.70%	89.07%
I=50, H=75	85.00%	87.90%	88.27%
I=75, H=25	87.77%	92.47%	92.47%
I=75, H=50	89.25%	92.47%	93.07%
I=75, H=75	88.15%	92.12%	92.82%

their classification rate over the above detailed database of images. The results of the comparison are reported in Table 1 for several model configurations[4]. Each model training has been repeated ten times, and the average performance of each model on ten testing runs has been calculated. The training algorithm used is backpropagation with momentum [26], combined with a 5-fold cross-validation procedure [29]. In each cross-validation partition of the database of faces, from all 10 images available per subject, 8 images of each 40 subjects (320 images) have been used for training purposes, and 2 images per subject have been used for testing (80 images).

Classification rates obtained for a classical MLP model with the standard training, the proposed aMLPs model with the standard training and the aMLPs trained with the proposed NCRT strategy are reported.

It can be observed that for all tested configurations, with different number of input and hidden neurons, the aMLPs model has obtained a better classification rate than a classical classifier for the database of images used. As could be expected, when more eigenfaces are included as inputs, better results can be achieved. However, the proposed aMLPs model with only 50 inputs can achieve almost 90% accuracy in recognition, while the classical MLP model needs more input eigenfaces to achieve a similar rate.

As can be seen, the NCRT strategy has improved significatively the classification rate of the aMLPs model when compared to a classical model. It can be said that the NCRT strategy may help in making a better use of the training patterns when there are scarce available data, balancing dynamically between class and no-class examples and improving the generalization capabilities. The second variant of the NCRT strategy, which uses as stopping criteria a minimun MSE, has been tried as well, obtaining very similar results (see Table 2) with a significant reduction in the training time.

[4] Using a 2.6 GHz Intel Core 2 Duo processor and 4 GB RAM memory.

Table 2. Comparison between a classical MLP classifier and the proposed aMLPs model. NCRT strategy for a fixed MSE: $MSE_{GOAL} = 2.5e{-}04$, $aMSE_{GOAL} = 1e{-}04$.

Neurons	MLP+basic	aMLPs+basic	aMLPs+NCRT
I=25, H=25	68.12%	74.27%	73.25%
I=25, H=50	71.80%	73.55%	73.97%
I=25, H=75	71.30%	72.57%	73.97%
I=50, H=25	85.50%	88.77%	89.45%
I=50, H=50	86.87%	88.67%	89.97%
I=50, H=75	87.02%	88.65%	90.52%
I=75, H=25	89.25%	92.52%	93.07%
I=75, H=50	90.35%	92.30%	93.17%
I=75, H=75	90.57%	92.15%	93.47%

Moreover, the sampling of the training space proposed by the NCRT strategy not only helps improving classification rates but also training time. In fact, the NCRT strategy used for the aMLPs model is al least 3 times faster than a classical training schema for achieving the same recognition rates. The novel NCRT strategy allows reaching better classification rate than the classical training, adjusting the model parameters, however, a significant minor number of times.

Figure 3 shows the relationship between training time and classification rate for aMLPs with $I = 50, H = 50$. It can be seen that when few negative examples are used, training time is high because it is hard for the model to learn from a few examples and also training epochs will be high. When a high number of negative examples is used, the training time is increased because the high number of no-classes examples. The intermediate situation can be found around 3 or 4 negative or no-class examples, which is where the highest classification rate and the minimun training time are reached. Furthermore, starting from 4 negative examples the training error rate is always 100%, giving a simple stopping criterion to maximize the generalization capabilities of the array.

In summary, additionaly to perform a better re-sampling of the no-class training space, in a reduced time, when compared to a classical MLP classifier, the recognition rates obtained with the variable training procedure are the highest.

6 Conclusions and future work

In this paper, an array of neural networks for face recognition and a novel no-class resampling training method for improving classification rates have been presented. The proposed model consists on one multilayer perceptron for each subject to recognize, performing the classification by the maximal output calculation among all the networks outputs.

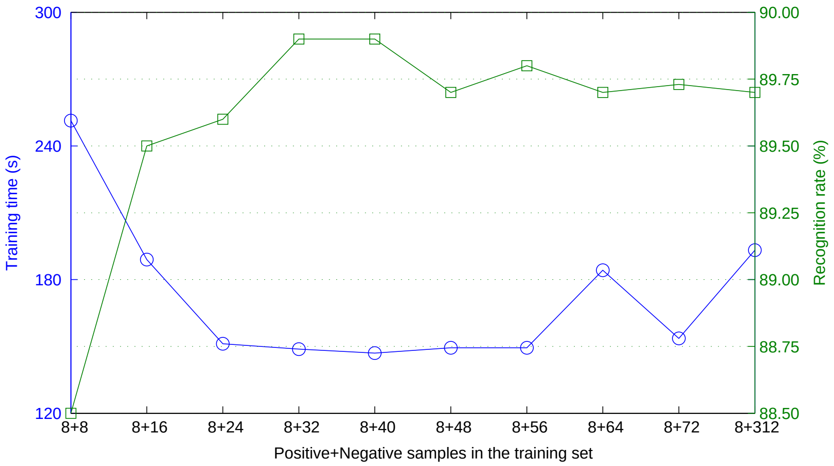

Fig. 3. Training time vs. Classification rates using NCRT strategy for aMLPs with $I = 50, H = 50, K = 40$.

The array of networks has been compared with the classic multilayer perceptron with one output unit for each subject. The obtained results showed an important improvement on the recognition. The novel NCRT strategy allows reaching the same classification rate as the aMLP with standard training but adjusting the model parameters a significant minor number of times.

Our future work includes using simpler network models for the aMLPs, for example, less hidden units in each network in the array, because in fact each one of them has to solve a simpler problem than the classical MLP classifier. We are also thinking in comparing the proposed NCRT algorithm against existing resampling or bootstrapping methods.

7 Acknowledgments

The authors wish to thank: the *Agencia Nacional de Promoción Científica y Tecnológica*, the *Universidad Nacional de Litoral (UNL)* (with CAI+D 012-72), the *Universidad Tecnológica Nacional (UTN-FRSF)* (with O/0105), the *UNL, UTN-FRSF and Universidad Nacional de Entre Ríos* (with PAE 37122, PAE-PICT-2007-00052), the *Consejo Nacional de Investigaciones Científicas y Técnicas* (CONICET) from Argentina, for their support.

References

1. Zhao, W., Chellappa, R., Phillips, P., Rosenfeld, A.: Face recognition: A literature survey. ACM Comput. Surv. **35**(4) (2003) 399–458
2. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3**(1) (1991)
3. Kirby, M., Sirovich, L.: Application of the Karhunen-Loeve procedure for the characterization of human faces. IEEE Transactions on Pattern Analysis and Machine Intelligence **12**(1) (1990) 103–108
4. Heisele, B., Ho, P., Wu, J., Poggio, T.: Face recognition: component-based versus global approaches. Computer Vision and Image Understanding **91** (2003) 6–21
5. Kong, S., Heo, J., Abidi, B., Palk, J., Abidi, M.: Recent advances in visual and infrared face recognition-a review. Computer Vision and Image Understanding **97**(1) (2005) 103–135
6. Eleyan, A., Demirel, H.: Pca and lda based neural networks for human face recognition. In Delac, K., Grgic, M., eds.: Face Recognition. I-Tech Education and Publishing, Vienna, Austria (2007) 93–106
7. Simon, J.: Resampling: The New Statistics, second edition. Academic Press (1997)
8. Bartlett, M., Movellan, J., Sejnowski, T.: Face Recognition by Independent Component Analysis. IEEE Trans. on Neural Networks **13**(6) (2002) 1450–1464
9. Jongsun, K., Jongmoo, C., Yi, J., Turk, M.: Effective representation using ICA for face recognition robust to local distortion and partial occlusion. IEEE Trans. on Pattern Analysis and Machine Intelligence **27**(12) (2005) 1977–1981
10. Martinez, A.M., Kak, A.C.: Pca versus lda. IEEE Trans. Pattern Anal. Mach. Intell. **23**(2) (2001) 228–233
11. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997) 711—720
12. Yang, M.H.: Kernel Eigenfaces vs. Kernel Fisherfaces: Face Recognition Using Kernel Methods. In: Proc. of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition. (2002) 215–220
13. Lu, J., Plataniotis, K., Venetsanopoulos, A.: Face recognition using Kernel Direct Discriminant Analysis algorithms. IEEE Trans. on Neural Networks **14**(1) (2003) 117–126
14. Yang, J., Frangi, A., Yang, J., Zhang, D., Jin, Z.: KPCA plus LDA: a complete kernel Fisher discriminant framework for feature extraction and recognition. IEEE Trans. on Pattern Analysis and Machine Intelligence **27**(2) (2005) 230–244
15. Yang, J., Zhang, D., Frangi, A., Yang, J.: Two-dimensional PCA: a new approach to appearance-based face representation and recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(1) (2004) 131–137
16. Khashman, A.: Face recognition using neural networks and pattern averaging. in Lecture Notes in Computer Science **3972**(1) (2006) 98–103
17. Sahoolizadeh, A.H., Heidari, B.Z., Dehghani, C.H.: A New Face Recognition Method using PCA, LDA and Neural Network. International Journal of Computer Science and Engineering **2**(4) (2008) 218–223
18. Bhowmik, M., Bhattacharjee, D., Nasipuri, M., Basu, D., Kundu, M.: Classification of polar-thermal eigenfaces using multilayer perceptron for human face recognition. (Dec. 2008) 1–6
19. Abate, A.F., Nappi, M., Riccio, D., Sabatino, G.: 2d and 3d face recognition: A survey. Pattern Recognition Letters **28**(14) (2007) 1885 – 1906

20. Zhang, D., Wangmeng, Z.: Computational intelligence-based biometric technologies. Computational Intelligence Magazine, IEEE **2**(2) (May 2007) 26–36
21. Rowley, H., Baluja, S., Kanade, T.: Neural network-based face detection. In: Proc. of the IEEE Computer Vision and Pattern Recognition. (1996) 203–208
22. Ebrahimpour, R., Kabir, E., Esteky, H., Yousefi, M.R.: View-independent face recognition with mixture of experts. Neurocomputing **71**(4-6) (2008) 1103 – 1107
23. C. Park, M. Ki, J.N., Paik, J.K.: Multimodal priority verification of face and speech using momentum back-propagation neural network. in Lecture Notes in Computer Science **3972**(1) (2006) 140–149
24. Bowyer, K.W., Chang, K., Flynn, P.: A survey of approaches and challenges in 3d and multi-modal 3d+2d face recognition. Computer Vision and Image Understanding **101**(1) (2006) 1 – 15
25. Fukunaga, K.: Introduction to Statistical Pattern Recognition, second edition. Academic Press (1990)
26. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning internal representations by error propagation. (1986) 318–362
27. Li, S.Z., Jain, A.K., eds.: Handbook of Face Recognition. Springer-Verlag (June 2004)
28. Sirovich, L., Kirby, M.: Low-dimensional procedure for the characterization of human faces. Journal of the Optical Society of America A **4**(3) (1987) 519–524
29. Haykin, S.: Neural Networks: A Comprehensive Fondation. Prentice Hall, New York (2002)