Normalized Radial Basis Function Networks and
Bilinear Discriminant Analysis for Face Recognition
Muriel Visani, Christophe Garcia, Jean-Michel Jolion

To cite this version:
Muriel Visani, Christophe Garcia, Jean-Michel Jolion. Normalized Radial Basis Function Networks
and Bilinear Discriminant Analysis for Face Recognition. IEEE International Conference on Advanced
Video and Signal based Surveillance (AVSS 2005), Sep 2005, Como, Italy. IEEE Computer Society,
pp.342-347, 2005. <hal-00452462>
Normalized Radial Basis Function Networks and Bilinear Discriminant Analysis for Face Recognition

Muriel Visani, Christophe Garcia
France Telecom Division R&D TECH / IRIS
4, rue du Clos Courtel
35512 Cesson-Sevigne, France
{muriel.visani, christophe.garcia}@rd.francetelecom.com

Jean-Michel Jolion
Laboratoire LIRIS, INSA Lyon
20, Avenue Albert Einstein
Villeurbanne, 69621 cedex, France
Jean-Michel.Jolion@insa-lyon.fr

Abstract

In this paper, we present a novel approach for face recognition using a new subspace method called Bilinear Discriminant Analysis (BDA), and Normalized Radial Basis Function Networks (NRBFNs). In a first step, BDA extracts the features that enhance separation between classes by using a generalized bilinear projection-based Fisher criterion, computed from image matrices directly. In a second step, the features are fed into a NRBFN that learns class conditional probabilities. This results in an efficient and computationally simple open-world identification process. Experimental results assess the performance and robustness of the proposed algorithm compared to other subspace methods combined with NRBFNs, in the presence of variations in head poses, facial expressions, and partial occlusions.

1. Introduction

During the last decade, automatic recognition of human faces has grown into a key technology, especially in the fields of multimedia indexing and security. In this context, the views of the face to recognize can differ drastically from the training set, especially in head poses, facial expressions and partial occlusions, which makes face recognition a difficult task.

Like any other pattern recognition task, face recognition can be basically defined as a two-step process: feature extraction and classification. In most surveillance applications, the features classification consists in 1) checking whether the corresponding face is registered in a database of known faces or not; and possibly 2) assigning an identity to that face. Since the seminal work of Sirovich and Kirby [13], statistical projection-based methods have been widely used for facial representation. In the Eigenfaces [14] and Fisherfaces [1] methods, a costly and potentially unstable statistical analysis is applied to the high-dimensional image vector space. To overcome these drawbacks, Yang et al. [18] proposed the Two Dimensional PCA (2DPCA) method, performing PCA using directly the face image matrices. It has been shown that 2DPCA is more efficient [18] and more robust [15] than the eigenfaces. Recently, we proposed the Two-Dimensional-Oriented Linear Discriminant Analysis (2DoLDA) approach [16], consisting in applying LDA on image matrices and outperforming the 2DPCA and Fisherfaces methods. The first contribution of this paper is a new feature extractor named Bilinear Discriminant Analysis (BDA), generalizing and outperforming 2DoLDA. This method is based on the optimization of a generalized Fisher criterion, relying on bilinear projections that are computed from image matrices directly.

The second contribution of this paper is the use of Normalized Radial Basis Function Networks (NRBFs) [5, 10] for the features classification. Indeed, as they generally provide better results in high dimensional problems [10, 12], and offer better generalization than the traditional RBFNs [2, 10, 6, 17], NRBFNs are particularly convenient for face recognition. Moreover, as they provide class conditional probabilities, NRBFNs can be easily used to check whether a person is registered or not.

The remainder of the paper is organized as follows. In section 2, we describe in details the principle and the algorithm of the proposed BDA method. In section 3, we present the classifier based on Normalized Radial Basis Functions Neural Networks. Then, in section 4, we present a series of two experiments, demonstrating the effectiveness and the robustness of the proposed algorithm and comparing its performances with respect to other subspace methods combined with NRBFNs. Finally, conclusions and closing remarks are drawn in section 5.

2 Bilinear Discriminant Analysis

The model is constructed from a training set containing $N$ face image matrices $X_i$, of size $h \times w$. The set of images corresponding to one person is called a class. Each class contains multiple views. The training set contains $k$ classes...
Let us consider two projection matrices \( Q \in \mathbb{R}^{k \times g} \) and \( P \in \mathbb{R}^{w \times g} \). Let us define the signature \( X_{i,j}^{Q,P} \) of the sample \( X_i \) by its bilinear projection onto \((Q, P)\):

\[
X_{i,j}^{Q,P} = Q^T X_i P
\]

We are searching for the optimal pair of matrices \((Q^*, P^*)\) maximizing the separation between signatures from different classes while minimizing the separation between signatures from the same class:

\[
(Q^*, P^*) = \arg\max_{Q, P} \frac{\sum_{j=1}^{n} n_j (X_{i,j}^{Q,P} - \bar{X}^Q)^T (X_{i,j}^{Q,P} - \bar{X}^Q)}{\sum_{j=1}^{n} \sum_{i \in \Omega_j} (X_{i,j}^{Q,P} - \bar{X}^Q)^T (X_{i,j}^{Q,P} - \bar{X}^Q)}
\]

(2)

where \( S_{Q}^{Q,P} \) and \( S_{P}^{Q,P} \) are respectively the within-class and between-class covariance matrices of the set \((X_{i,j}^{Q,P})_{i \in \{1, \ldots, N\}}\). \( X_{i,j}^{Q,P} \) is the mean of all the projected samples belonging to class \( j \), and \( \bar{X}^{Q,P} \) is the mean of all the projected samples of the training set.

The objective function given in equation (3) is bi-quadratic and has no analytical solution. We therefore propose an iterative procedure that we call Bilinear Discriminant Analysis. Let us expand the objective function (3):

\[
(Q^*, P^*) = \arg\max_{Q, P} \frac{\sum_{j=1}^{n} n_j (X_{i,j}^{Q,P} - \bar{X}^Q)^T Q^T P (X_{i,j}^{Q,P} - \bar{X}^Q)}{\sum_{j=1}^{n} \sum_{i \in \Omega_j} (X_{i,j}^{Q,P} - \bar{X}^Q)^T Q^T P (X_{i,j}^{Q,P} - \bar{X}^Q)}
\]

(4)

Thus, for any fixed \( Q \in \mathbb{R}^{k \times g} \), the objective function (3) can be rewritten:

\[
P^* = \arg\max_{P \in \mathbb{R}^{w \times g}} \frac{\sum_{j=1}^{n} n_j (X_{i,j}^{Q,P} - \bar{X}^Q)^T Q^T P (X_{i,j}^{Q,P} - \bar{X}^Q)}{\sum_{j=1}^{n} \sum_{i \in \Omega_j} (X_{i,j}^{Q,P} - \bar{X}^Q)^T Q^T P (X_{i,j}^{Q,P} - \bar{X}^Q)}
\]

(5)

\[
P^* = \arg\max_{P \in \mathbb{R}^{w \times g}} \frac{|P^T S_{Q}^{Q,P} P|}{|P|}
\]

(6)

with \( S_{Q}^{Q} \) and \( S_{Q}^{P} \) being respectively the generalized within-class covariance matrix and the generalized between-class covariance matrix of the set \((X_{i,j}^{Q})_{i \in \{1, \ldots, N\}}\), where \( X_i^Q = Q^T X_i \). Therefore the columns of the matrix \( P^* \) are the \( g \) eigenvectors of \( S_{Q}^{Q,P} \) with largest eigenvalues.

For any fixed \( P \in \mathbb{R}^{w \times g} \), using equation (7) the objective function (3) can be rewritten \( Q^* = \arg\max_{Q \in \mathbb{R}^{k \times g}} \frac{|Q^T S_{P}^{Q,P} Q|}{|Q|} \), \( \Sigma_{w} \), \( \Sigma_{b}^{Q} \) and \( \Sigma_{b}^{P} \) being the generalized within-class and between-class covariance matrices of the set \((X_{i,j}^{Q,P})_{i \in \{1, \ldots, N\}}\), where \( X_i^P = X_i \cdot P \). Therefore, the columns of \( Q^* \) are the \( g \) eigenvectors of \( (\Sigma_{w})^{-1} \Sigma_{b}^{Q} \) with largest eigenvalues. If the set \( P = I_w \), the matrix \( Q^* \) is the projection matrix of Column-Oriented Linear Discriminant Analysis (CoLDA) [16].

We can note that BDA leads to a significant reduction in the dimensionality of the signatures compared to 2DPCA and 2DoLDA: the size of a signature using BDA is \( g^2 \), versus \( h \cdot g \) for RolDA and 2D PCA, and \( w \cdot g \) for CoLDA.

The algorithm of the BDA approach is as follows: let us initialize \( P_0 = I_w \) and \( \alpha_0 = 0 \). The number of feature vectors \( g \) is fixed (see section 4).

**While** \( \alpha_t < \tau \)

1. For \( l \in \{1, \ldots, N\} \), compute \( X_t^{P_l} = X_t \cdot P_l \).
2. Compute \( \Sigma_{w}^{P_l}, \Sigma_{b}^{P_l} \) and \( (\Sigma_{w}^{P_l})^{-1} \cdot \Sigma_{b}^{P_l} \).
3. Compute \( Q_t \), whose columns are the first \( g \) eigenvectors of \( (\Sigma_{b}^{P_l})^{-1} \cdot \Sigma_{b}^{P_l} \).
4. For \( l \in \{1, \ldots, N\} \), compute \( X_t^{Q_l} = (Q_l)^T \cdot X_t \).
5. Compute \( S_{w}^{Q_l}, S_{b}^{Q_l} \), and \( (S_{w}^{Q_l})^{-1} \cdot S_{b}^{Q_l} \).
6. Compute \( P_t \), whose columns are the first \( g \) eigenvectors of \( (S_{w}^{Q_l})^{-1} \cdot S_{b}^{Q_l} \).
7. Compute \( \alpha_t = \sqrt{\|P_t - P_{t-1}\|^2 + \|Q_t - Q_{t-1}\|^2} \).

The parameter \( \tau \) can be determined empirically. However, experimental results have shown that after one iteration the recognition results are satisfying. Therefore, in the following, we will use the preceding algorithm with only one iteration, freeing us from determining \( \tau \).

### 3. Normalized RBF Network

Radial Basis Function Networks can be used for classification as a two-layer neural network implementing a mapping function \( \mathbb{R}^n \rightarrow \mathbb{R}^k \), where \( n \) is the size of the input signal and \( k \) is the number of classes. The two layers are: the RBF layer, and the output layer. For any input signal \( X_t \), the Euclidean distance to the center of each of the \( r \) RBF units is computed, and then an activation function \( R \) is applied. Although there are many possible activation functions, the Gaussian function is preferred for high-dimensional data [12]: for each input observation \( X_t \), the output of the \( i \)-th RBF is given by:

\[
R_{i,t} = e^{-\frac{1}{2\sigma_i^2}(X_t - C_i)^T(X_t - C_i)}
\]

(8)
where $C_i$ and $\sigma_i$ are respectively the center and the width of the $i^{th}$ RBF. The $k$ output units are linear: if $W_0$ denotes the bias of the system, the response of the $j^{th}$ output is:

$$Y_{j,t} = W_0 + \sum_{i=1}^r W_{ji} \cdot R_{i,t}$$

(9)

The system constructed from (9) can be rewritten:

$$Y = W \cdot R$$

(10)

where the elements of the matrix $W \in \mathbb{R}^{g \times r+1}$ are the $W_{ji}$ and, $\forall j \in \{1 \ldots k\}, W_{j0} = W_0$. The matrix $R \in \mathbb{R}^{r+1 \times N}$ contains the elements $R_{i,t}$, and $\forall t \in \{1 \ldots N\}, R_{0t} = 1$. The elements of $Y \in \mathbb{R}^{k \times N}$ are the $Y_{j,t}$.

The specificity of the Normalized RBF Network, which architecture is illustrated in Figure 1, is that the output of each RBF unit is normalized by the total activity of the RBF layer. The outputs of the NRBFs are normalized by the total activity of the RBF layer, and therefore if we consider that $R_{i,t} = P[i/X_t]$ and using Bayes’ theorem, the NRBF outputs are the posterior probabilities of $X_t$ belonging to the $i^{th}$ NRBF hypersphere:

$$P[i/X_t] = \frac{R_{i,t}}{\sum_{i=0}^r R_{i,t}}$$

(11)

The $j^{th}$ output of the overall network can be viewed as the posterior probability of $X_t$ belonging to class $\Omega_j$:

$$Y_{j,t} = P[\Omega_j/X_t] = \sum_{i=0}^r W_{ji} \cdot P[i/X_t]$$

(12)

and the weights can be expressed as: $W_{ji} = P[\Omega_j/i]$. The system constructed from (9) can be rewritten:

$$Y = W \cdot R$$

(10)

where the elements of the matrix $W \in \mathbb{R}^{g \times r+1}$ are the $W_{ji}$ and, $\forall j \in \{1 \ldots k\}, W_{j0} = W_0$. The matrix $R \in \mathbb{R}^{r+1 \times N}$ contains the elements $R_{i,t}$, and $\forall t \in \{1 \ldots N\}, R_{0t} = 1$. The elements of $Y \in \mathbb{R}^{k \times N}$ are the $Y_{j,t}$.

The specificity of the Normalized RBF Network, which architecture is illustrated in Figure 1, is that the output of each RBF unit is normalized by the total activity of the RBF layer. The outputs of the NRBFs are normalized by the total activity of the RBF layer, and therefore if we consider that $R_{i,t} = P[i/X_t]$ and using Bayes’ theorem, the NRBF outputs are the posterior probabilities of $X_t$ belonging to the $i^{th}$ NRBF hypersphere:

$$P[i/X_t] = \frac{R_{i,t}}{\sum_{i=0}^r R_{i,t}}$$

(11)

The $j^{th}$ output of the overall network can be viewed as the posterior probability of $X_t$ belonging to class $\Omega_j$:

$$Y_{j,t} = P[\Omega_j/X_t] = \sum_{i=0}^r W_{ji} \cdot P[i/X_t]$$

(12)

and the weights can be expressed as: $W_{ji} = P[\Omega_j/i]$. The proposed adjustment, inspired from [6], is:

$$\sigma_i^W = \frac{d_{i}^{W}}{\sqrt{\log(\beta)}}$$

$$\sigma_i^B = \mu \cdot d_{i}^B$$

(13)

where the parameter $\mu$ can be estimated as follows:

$$\mu \approx \frac{\sum_{j=1}^k \sigma_j^W}{\sum_{j=1}^k d_{j}^B}$$

(15)

The parameter $\beta \in [0.5, 1]$ is determined by the relative positions of the classes: the more largely scattered are the data, the smaller should be $\beta$.

**Hybrid Learning Algorithm.** In hybrid learning, the adjustment of the centers and widths of the RBFs is a non-linear process while the weight learning is a linear one. It combines the gradient paradigm and the Linear Least Squares (LLS) optimization method. If we consider, for each training sample $X_t$, the pair of obtained and desired outputs $(Y_{t}, T_{t})$ (the matrix $T$ contains only ‘0’s and ‘1’s), we can jointly optimize the set of parameters of the RBFs $\{C_i, \sigma_i\}_{i=1}^r$ and the weights $W$ by minimizing the following cost function:

$$E = \sum_{t=1}^N E^t = \frac{1}{2} \sum_{t=1}^N (T_{t} - Y_{t})^T (T_{t} - Y_{t})$$

(16)
At each epoch \(t\), the centers and widths are adjusted for each sample \(X_i\):

\[
C_i = C_i - \xi_c \Delta C_i^t \quad \text{(17)}
\]

\[
\sigma_i = \sigma_i - \xi_\sigma \Delta \sigma_i^t \quad \text{(18)}
\]

where \(\Delta C_i^t\) and \(\Delta \sigma_i^t\), whose associated learning rates are respectively \(\xi_c\) and \(\xi_\sigma\), can be computed as follows:

\[
\forall \ m \in \{1 \ldots n\}, \quad \Delta C_i^t(m) = \frac{\partial E^t}{\partial C_{i,m}} = -\frac{(x_{i,m} - c_{i,m})^2}{\sigma_i^2(\sum_{i=1}^n R_{i,i})^2} \cdot R_{i,i} 
\]

\[
\cdot \left(\sum_{i=1}^r R_{i,i} - R_{i,i}\right) \cdot \frac{k}{j=1} W_{j,i} (T_{j,i} - Y_{j,i}) \quad \text{(19)}
\]

\[
\text{and} \quad \Delta \sigma_i = \frac{\partial E^t}{\partial \sigma_i} = -\frac{\sum_{j=1}^n (x_{j,m} - c_{j,m})^2}{\sum_{j=1}^n (x_{j,m} - c_{i,m})^2} \cdot R_{i,i} 
\]

\[
\cdot \left(\sum_{i=1}^r R_{i,i} - R_{i,i}\right) \cdot \frac{k}{j=1} W_{j,i} (T_{j,i} - Y_{j,i}) \quad \text{(20)}
\]

Once the parameters \([C_i, \sigma_i]\) have been settled, the NRBF outputs \(P[j_i/X_i]\) can be computed from (11). Then, the weight matrix \(W\) can be determined by applying the LLS method based on the pairs \((Y_i, T_i)_{i=1\ldots n}\):

\[
W = TR^T (RRT)^{-1} \quad \text{(21)}
\]

**Checking whether a Face is Registered in the Database.**

For most surveillance system, an alert should be given only when a registered face is detected. Let us denote \(j^t\) the most probable class. We propose the following criterion:

\[
P[j^t/X_t] > \tau_t \quad \text{(22)}
\]

where \(\tau_t = 0.25\) and the false rejections are considered as misclassifications.

The first experiment is designed to evaluate these classifiers when the persons in the test set are registered in the training set, but with strong dissimilarities between the two sets. The first test set contains strong facial expression changes (see Figure 2(b)). The second test set contains one top-right head pose view per person (see Figure 2(c)). The third test set (see Figure 2(d)) contains one digitally modified image per person, where a black strip of width 25 pixels, simulating a scarf, has been added. The original image does not belong to the training set.

The experimental results show that the use of the NRBF Network does not improve the recognition results, compared to the Nearest Neighbour Classifier (NNC), in the PCA+LDA subspace. Moreover, the performances of PCA+LDA+NRBFN are highly dependent on the number of RBFs, and on their initialization; that is why they are not presented in this paper. Table 1 provides a comparison of the performances of PCA, PCA+LDA, 2DPCA, the two versions of 2DoLDA (namely CoLDA and RoLDA) and BDA, using NNC. From this table we can see that, as highlighted in [15], 2DPCA outperforms PCA, especially when dealing with partial occlusions. Table 1 also highlights the fact that BDA outperforms the other projection-based methods, when using NNC.

The Mean Square Errors (MSE) between the obtained and desired outputs \(Y\) and \(T\), using BDA+NRBFN and PCA+NRBFN, are given in Figure 3(a–b). From Figure 3(a) we can see that the learning errors are initially comparable for the two methods; both decrease when the number of epochs increase, but the MSE for PCA+NRBFN decreases much more slowly than the MSE for BDA+NRBFN. Figure 3(b) shows that the MSE for PCA+NRBFN slightly increases with the number of epochs over the two first test sets, while the MSE for BDA+NRBFN significantly decreases. Therefore, BDA+NRBFN seems to have a better generalization power than PCA+NRBFN. The compared recognition rates on the three test sets, given in Figure 4(a–b), verifies that assumption. It should be noted that, for both methods and after a sufficient number of iterations, the false rejection rate is null among the three test sets, ie all the faces are correctly classified as registered in the training data.

When evaluating BDA+NRBFN on the “pose” test set, one misclassification and two false rejections occur; however, after 4800 epochs the recognition rate reaches 100%. Let us denote \(P[\text{target}]\) the probability of belonging to the target class, and \(P[\text{wrong}]\) the probability of belonging to the wrong class with highest associated probability. Figure 5(a) shows that after 4800 iterations, \(P[\text{target}] > P[\text{wrong}]\) and \(P[\text{target}] > \tau_t\), therefore the sample is recognized. Figure 5(b–c) illustrates the two false rejections: in both cases,
after respectively 2450 and 1900 epochs, $P[\text{target}]$ reaches $\tau_1 = 0.25$ while $P[\text{wrong}]$ remains inferior to 0.25 and then the corresponding sample is recognized.

The first experiment results show 1) the superiority of BDA+NRBFN over PCA+NRBFN and PCA+LDA+NNC (and, consequently, over PCA+LDA+NRBFN) in the presence of variations in facial expression, head pose, and facial occlusion and 2) that, after a sufficient number of epochs, all the registered faces are correctly classified as registered.

The second experiment aims at testing whether the proposed classifier correctly classifies the unregistered people as unknown. The training set and the parameters of the model are the same as in the first experiment. The test set (see Figure 2(c)) contains one image of each of 15 unregistered persons, under the same illumination conditions as in the training set and near-frontal head poses. Figure 4(c) shows that BDA+NRBFN generates less false alarms than PCA+NRBFN, as it correctly classifies 93.3% of the faces as unregistered, versus 86.7% for PCA+NRBFN.

5. Conclusion

In this paper, we have presented a novel approach for face recognition, combining a new feature extractor named Bi-linear Discriminant Analysis and Normalized Radial Basis Function Networks, that to the best of our knowledge have never been applied to face recognition. This method, by providing class conditional probabilities, can be used to check whether a person is registered in the training data and, possibly, assign an identity to that person. The experimental results assess the effectiveness and robustness of the proposed approach and show its superiority over the usual projection-based methods, combined with Normalized Radial Basis Function Networks and the Nearest Neighbour rule.

References


Figure 1: Architecture of a NRBF Network.

Table 1: Compared Recognition Rates of PCA, PCA+LDA, 2DPCA, CoLDA, RoLDA and BDA using the Nearest Neighbour Rule.

<table>
<thead>
<tr>
<th></th>
<th>PCA</th>
<th>PCA+LDA</th>
<th>2DPCA</th>
<th>CoLDA</th>
<th>RoLDA</th>
<th>BDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>'pose'</td>
<td>96.7%</td>
<td>96.3%</td>
<td>96.7%</td>
<td>96.7%</td>
<td>98.3%</td>
<td>100%</td>
</tr>
<tr>
<td>'expression'</td>
<td>95%</td>
<td>93.3%</td>
<td>95%</td>
<td>96.7%</td>
<td>96.7%</td>
<td></td>
</tr>
<tr>
<td>'occlusion'</td>
<td>13.3%</td>
<td>50%</td>
<td>55%</td>
<td>70%</td>
<td>53.3%</td>
<td>70%</td>
</tr>
</tbody>
</table>

Figure 2: (a): training set; (b) "expression" test set; (c) "pose" test set; (d) "occlusion" test set; (e): set of unregistered people.

Figure 3*: Compared Mean Square Errors of BDA+NRBFN (solid line) and PCA+NRBFN (dashed line) computed (a): from the training set; b): from the first two test sets, when varying the number of epochs.

Figure 4*: Evolution of the recognition rates of BDA+NRBFN and PCA+NRBFN computed: (a) from the first two test sets and (b) from the "occlusion" set; (c): rejection rates computed from the set of unregistered people.

Figure 5*: Class conditional probabilities of the target class ("P[target]") and of the most probable wrong class ("P[wrong]"), for (a) the misclassification; (b-c) the two false rejections, when varying the number of epochs.

*Note that the vertical scales vary from plot to plot in this Figure.