UNIVERSIDAD REY JUAN CARLOS
Servicio de Publicaciones

MORE IN-DEPTH ANALYSIS OF 2D LDA

WORKING PAPERS 2006/03

Hans Marcus Krüger
Licesio J. Rodríguez-Aragón
Cristina Conde
Ángel Serrano
Enrique Cabello

Febrero, 2006

More in-depth Analysis of 2D LDA

Hans Marcus Krüger, Licesio J. Rodríguez-Aragón,
Cristina Conde, Ángel Serrano and Enrique Cabello
hans@wh2.tu-dresden.de,licesio.rodriguez.aragon@urjc.es
cristina.conde@urjc.es,angel.serrano@urjc.es,
enrique.cabello@urjc.es
http://frav.escet.urjc.es

May, 2005

Abstract

Spatial dimension reduction called Two Dimensional LDA method has recently been presented. The application of this variation of traditional LDA considers images as 2D matrices instead of 1D vectors as PCA and LDA, traditional dimension reduction, methods have been using.

The spatial approaches are more reliable for the purpose of face verification but deeper work has to be done to understand the way the information is extracted to achieve a more accurate verification. The goal of these studies is to analyze more in depth how Two Dimensional LDA behaves in face recognition.

1 Introduction

This studies have been realized during the visit of Hans Marcus Krüger to the *Face Recognition and Artificial Vision group* (FRAV) at *Universidad Rey Juan Carlos*, Madrid, Spain, supported by an Erasmus grant.

1.1 Who is FRAV?

The *Face Recognition and Artificial Vision Group* (FRAV, http://frav.escet.urjc.es/) at *Universidad Rey Juan Carlos*, Madrid, Spain is a research group conducting studies in the fields of face recognition and traffic safety based on automated video surveillance. Among other projects fined by local organizations and the European Union, we conduct studies in the fields of 2D and 3D face recognition.

1.2 What this report wants to show?

This report is intended to illustrate the studies done on 2D Linear Discriminant Analysis (2DLDA), realized by the author of this report during his stay at the University. The studies where realized in a 6 months term and provide a little-more-in-depth view of 2DLDA.

The paper will make a short introduction to the way face recognition is realized using PCA, LDA and 2DLDA. It will also discuss in a little more detailed way how *Principal Component Analysis* (PCA) works and what its strengths and limitations are. This part is a good starting point to take a look at *Linear Discriminant Analysis* (LDA) and 2DLDA.

The report will try to show why 2DLDA is said to perform better than PCA or LDA but will also try to show some of its limitations. In the end, it will make some more in-depth analysis on the results that where produced with 2DLDA.

Improving security and developing new smart environments are some of the key points in which biometry plays a most relevant role. Recent studies [1] have shown that technology is in very early stages of development to perform surveillance tasks at critical locations. However, simulations or real tests are crucial to obtain the required feedback in order to improve in the right direction. Dimensionality reduction is an important and necessary preprocessing of multidimensional data, as face images.

We expect, that the image is given to us already in a defined format, pre processed, and we also will not worry about how our results are actually matched to a given person. We will focus or work on describing the way the methods extract the required information to be able to perform a face verification task.

2 Face recognition

Face recognition can be realized in very different ways. Some try to find characteristic points in the face, like nose tip, ears and eyebrows create a metric map over this point and try to perform some classification on the distances and angles between them. Others segment the whole face (in nose, left eye, right eye etc.), to do some kind of classification on them and then try to recombine those results to obtain a final answer.

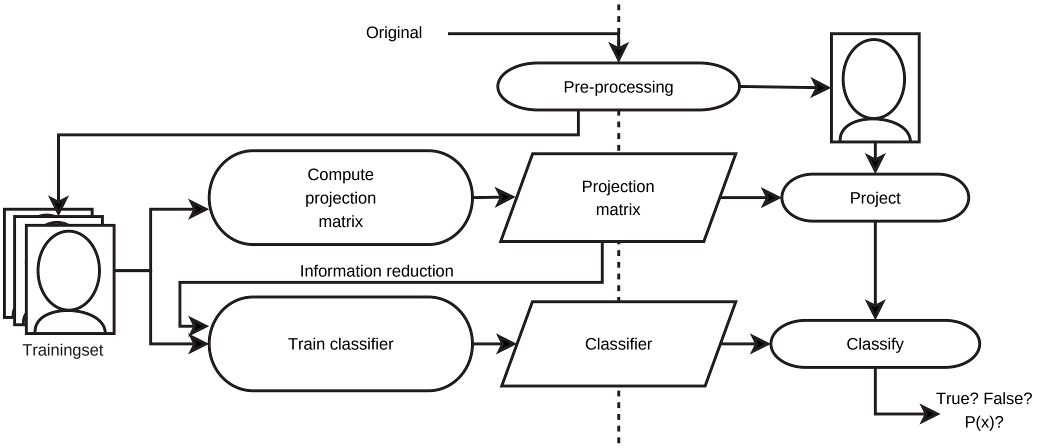


Figure 1: System Overview: its more significant parts are a preprocessing module, a feature extraction process and the final decision is taken by a previously trained classifier.

Traditional biometric systems are based on applying statistical methods over a face image as a whole. Only previous image preprocessing like equalizing and cropping is done.

Although 2DLDA and all other algorithms shown in this work do work for any type of feature recognition, this work will study all algorithms only in the context of face recognition. Therefore we will make an introduction to face recognition as is performed in the systems using the methods present in this report.

Figure 1 shows a schematic overview of a generic system. On the right the authentication is performed. First the image is projected by a previously calculated matrix to a new dimensional space (feature extraction). This new vector, that contains much less information can then be insert into a classification system to make the decision, if a certain image belongs to an entry in our database or not.

To build up this classification system some computation is required in advance, which is illustrated on the left side. Given a set of images, some restrains apply to the set, depending on the algorithms used. For example, all systems based on LDA will require a set of multiple images for every subject in the database.

Those images are used to create a projection matrix that fit certain *needs*, which will be specified further on. Different algorithms (PCA, LDA, ...) have different characteristics and will produce matrices that do fulfill our needs in different ways. Specifying this *needs* is quite simple: We need a transformation, which projects the Face Image to a new vector or matrix ideally only large enough to contain the information needed to distinguish all possible faces from each other with higher accuracy as possible. Still no proof of the existence of such a matrix exists.

PCA and LDA are statistical operations. Therefore they will produce results that can be described with the statistical quantities. The two most important ones are *variance* and *average*. System based on PCA and LDA will only allow statistical analysis. This does not mean that this approach is deemed to fail. It is a matter of how good the analysis of the images is.

It is expected that this new multi dimensional vector space created by the transformation might allow better results than doing a byte-by-byte comparison of a image as the algorithm is expected to extract features of the images that will be used by a classifier. So the seek for a new representation of the image is not only to reduce information but also to improve the classification itself.

Once this transformation matrix is known, the training images can be projected and fed to the training process of the classification system. Classification can be realized through neuronal networks, super vector machines or classification by nearest neighbour (KNN). It is beyond the scope of this paper to discuss different classification systems. For the tests realized in the end of this work, classification by nearest neighbour was used.

2.1 Images Database: FRAV2D

The Face Recognition and Artificial Vision group (FRAV) at the Universidad Rey Juan Carlos, has collected a quite complete set of facial images for 109 subjects. All the images have been taken under controlled conditions of pose and illumination. 32 images were taken of each subject, being 12 frontal, 4 preforming a 15° rotation, 4 performing a 30° rotation, 4 with cenital instead of diffuse illumination, 4 performing different gestures and 4 occluding parts of the face. A partial group of this database is freely available for research purposes.

The images are colored and of size 240 × 320 pixels with homogeneous background color. A window of size 140 × 130 pixels containing the most meaningful part of the face, has been automatically selected in every image and stored in equalized gray scale. That is the information that will be analyzed through the dimension reduction and classification methods.

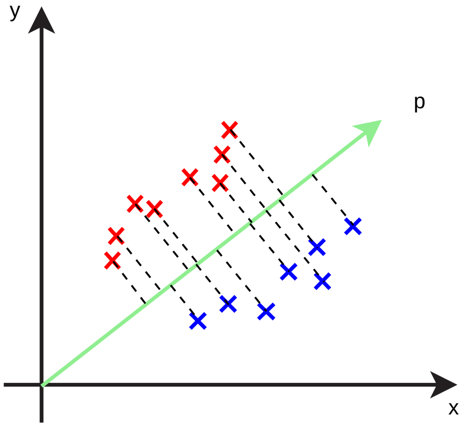


Figure 2: Graphical example of PCA. The first principal component p is selected to maximice the variance of the data.

3 Principal Component Analysis

Principal component analysis (PCA) is a classical method in the field of statistics. The main goal is to find a linear combination of the variables, described as a vector m containing the scalars of this linear combination, for which the projection of the original data onto this new variable have the maximal variance. In the same way it can be generalized to multiple vectors: the best linear combination of variables to retain the higher amount of variance, second best and so on. This way we obtain n vectors m_i formed by the corresponding scalars. This vectors can be grouped together into a matrix $M = |m_1, m_2, \ldots m_n|$ obtaining what is known as the projection matrix.

Figuratively speaking, a cloud of data will be projected onto a new axe obtained as a linear combination of the two original axes. This new direction obtained maximices the variance of the data. Figure 2 shows a initial probe. It consists of two clouds of eight points, each one belonging to the group A or B.

3.1 The Math

Given a set of images $I_1, I_2, \ldots, I_N$ of height h and width w, PCA considers the images as 1D vectors in a $h \cdot w$ dimensional space. The facial images are projected onto the eigenspace spanned by the leading orthornormal eigenvectors, those of higher eigenvalue, from the sample covariance matrix of the training images. Once the set of vectors has been centered, the sample covariance matrix is calculated, resulting a matrix of dimension $h \cdot w \times h \cdot w$.

$$C^* = |I_1, I_2, \ldots, I_N|$$

$$C = cov(C^*) = C^* \times transpose(C^*)$$

It is widely known that if $N \ll h \cdot w$, there is no need to obtain the eigenvalue decomposition of this matrix, because only N eigenvectors will have a non zero associated eigenvalue [3].

Over this new matrix, eigenvalue decomposition is performed. It will return a set of eigenvalues e_i and eigenvectors v_i. The number n of e_i and v_i is the rank of the matrix. If C is non singular and square, n equals to the number of columns/rows, $h \cdot w$, in the matrix.

All vectors v_i of the eigenvalue decomposition are orthogonal from each other. This means, that they span a new vector space. Even more, the absolute value of e_i do also contain information about how important the corresponding vector v_i is to represent the probe. Vectors with corresponding big absolute eigenvalues contain more information. In other words, if the projection is done onto a vector with a high eigenvalue, the variance will be more expressive, than if a vector with less absolute eigenvalue is chosen. In figure 2 p is the vector that maximizes the variance.

In the next step the m best eigenvectors are grouped to form the projection matrix M. This matrix is used to project each image I into the new vector space.

$$transpose(s_{(m,1)}) = transpose(I_{(h \cdot w,1)}) \times M_{(h \cdot w,m)}$$

This process can be inverted. From the projected vector $s_{(m,1)}$ and trough the projection matrix M a reconstructed image can be obtained. This can be archived by

$$\tilde{r}_{(h \cdot w,1)} = M_{(h \cdot w,m)} \times s_{(m,1)}$$

Each eigenvector e_i has dimension $h \cdot w$ and can be represented as an image of height h and width w. What is represented in these images are the scalars that form the linear combination of the principal component. High values are assigned to high intensities of white, while low values are assigned to high intensities of black. Therefore white pixels represent variables with a high weight in the projected image. This images are called *eigenfaces*. Also reconstructed images $\tilde{r}$ can be represented as images and the features extracted are then visualized. Figure 3 shows how they look like.

In face recognition this lossless bidirectional transformation is not necessarily needed. In fact, the aim is to find a new representation of the image

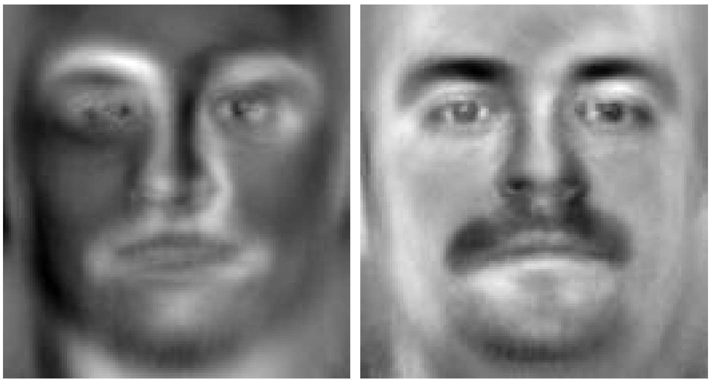


Figure 3: Left: Representation of the first eigenface, eyes contour, nose and mouth are the most significant features in the new principal component. Right: Reconstruction of a projected image.

which should contain as much information as possible for performing an accurate classification.

The problem to be solved is which eigenvectors to take. How many principal components are needed to perform a correct classification. Eigenvalues can be sorted in decreasing value. As seen before the eigenvector corresponding with the higher eigenvalue represents the linear combination of variables that retain the highest amount of variance. Eigenvectors can then graphically represented as in Figure 4 which shows such a graph of 130 eigenvalues. At this is point, the number of principal components m can be defined. If the projections s should retain $v = 75\%$ of the information (variance), M needs to contain the m-most representative vectors till the corresponding eigenvalues e_i sum up 75% of the sum of all e_i:

$$v = \frac{\sum_1^m e_i}{\sum e_i} \cdot 100$$

3.2 Draw backs

There are some points on PCA that should be mentioned. First of all, when C^* is created the probes from A and B are concatenated discarding the information to what group a entry belongs. Form there on the whole process does not know anything about different groups any more. This arises a lot of problems.

1. If one group has much more entries, it will influence the generation of the eigen vectors in the way, that it will dominated in the process by *bending* the vector towards it' s average.

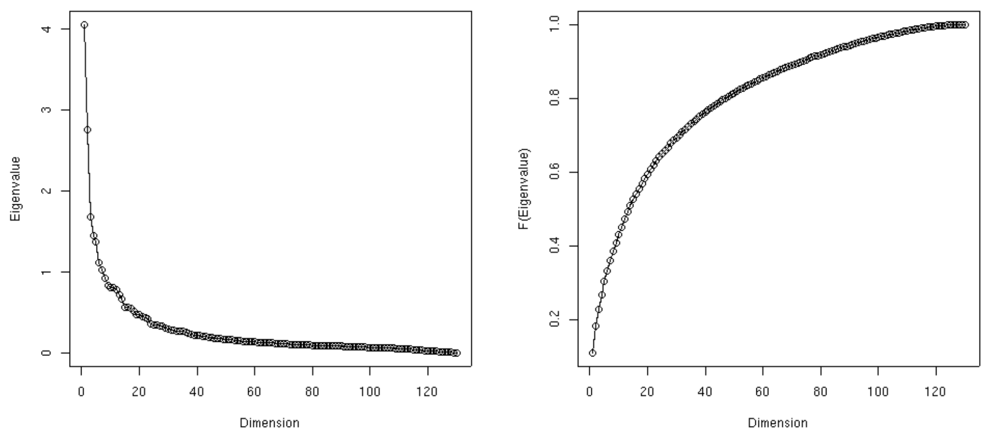


Figure 4: Eigen values

2. Even worse —the process seeks for a maximal overall scatter. But this scatter is not necessarily good for classification. In the example (figure 2) p is not suitabel at all to realize classification, as both clouds are projected onto the same region of the vector.

Other problems arise, when it comes to working with images. An image is a two dimensional numerical array with h rows and w columns. To be able to apply PCA on images they have to be introduced in a matrix C^*. To archive this, the images are transformed to column vectors by concatenating the lines to a large vector of size wh. In this transformation the two dimensional neighbourhood of the pixels is destroyed. This certainly is a loss of information.

One even bigger problem is the size of the matrix C^*. It will have wh rows and n columns, being n the number of images we include. Therefor C will have the dimension of $wh \times wh$ which is a very big number. For small images this might work, but as quality increase resources for doing the math will become more scarce rapidly.

3.3 PCA by M. Turk

1991 M. Turk [3] came up with a solution to avoid big matrices. Instead of computing the covariance matrix C over the variables of the image he suggested transposing the whole system and compute the covariance over the individuals:

$$C' = transpose(C^*) \times C^*$$

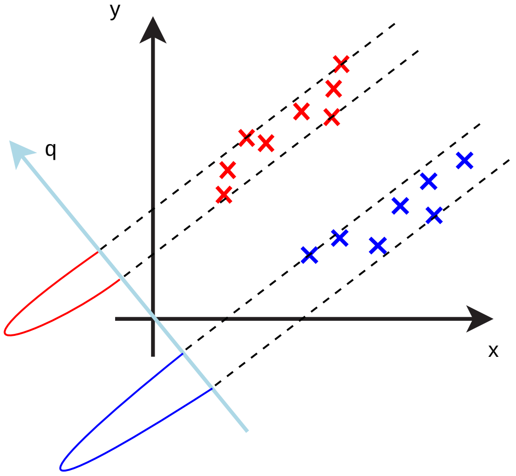


Figure 5: LDA

With this trick images with more resolution could be used. Also, the size of the matrix grows to the square of the number of individuals, this number commonly is much smaller than wh. Never the less, the obtained matrix $M'_{(n,n')}$ is not suitable for projecting the images, as the dimensions do not fit. This matrix will have to be transformed first.

$$M_{(hw,w')} = C^*_{(wh,n)} \times M'_{(n,n')}$$

This approach will work as log $n^2 \leq m$ holds true.

4 Linear Discriminant Analysis

Linear discrimant analysis (LDA) is similar to PCA but instead of maximising the overall scatter it maximises the between scatter and minimises the within scatter. Figure 5 illustrates this. Red and blue probes are projected so, that they do not overlap and the average of both clouds are far away from each other. Good classification is expected from this projection.

LDA does know the concept of different groups. The between scatter is the summed up difference from each groups average to the overall average. Within scatter is obtained by summing up the distances of each entry to it corresponding average.

The search for the new base vectors is done by maximising the *fish criterion* $J(X)$.

$$x = arg\max_x J(x) = arg\max_x \frac{x^T S_B x}{x^T S_W x}$$

It reads: We search an optimal vector x so that $J(x)$ is maximal. The clouds will be projected in a very dense way (small $x^T S_W x$) and spread from each other (big $x^T S_B x$).

S_B and S_W are the between and the within scatter matrices. Images are processed as in PCA by converting them to single row vectors.

In LDA the solution is found by transforming the above equation to a general eigen value problem. This approach is only valid, if S_W can be inverted an so, if it is non-singular.

In the next section LDA will be described more in depth in conjunction with 2D LDA.

5 2DLDA

2004 Li and Yuan [2] proposed a new way for computing the projection matrix. Up till now the images were transformed into a single column vector. They claim, that with this new proposal, 2D neighbourhood is maintained for the pixels.

As already seen in section 4 a fisher criteria

$$J(x) = \frac{P_B}{P_W}$$

has to be defined, which is to be maximised.

P_B should denote the *between scatter* and P_W the *within scatter*. If P_B is maximal, and P_W minimal, $J(x)$ will be maximal too.

The scatter matrices are computed in the following way:

$$\begin{aligned}
P_B &= x^T S_B x \\
P_W &= x^T S_W x \\
S_B &= \sum_{i=1}^{L} N_i(\overline{A_i} - \overline{A})^T(\overline{A_i} - \overline{A}) && (1) \\
S_W &= \sum_{i=1}^{L} \sum_{A_k \in T_i} (A_k - \overline{A_i})^T(A_k - \overline{A_i}) && (2)
\end{aligned}$$

(1) reads: For all L existend groups, sum up the difference between the group average $\overline{A_i}$ and the overall average $\overline{A}$. (2) reads: For each group T_i with $1 \leq i \leq L$, sum up the difference of each image A_k to the group average $\overline{A_i}$.

In the next step a vector x is searched that satisfy the fisher criteria defined above.

$$x = \underset{x}{argmax} J(x) = \underset{x}{argmax} \frac{x^T S_B x}{x^T S_W x}$$

If S_W is non-singular, the solution is obtained by solving the generalised eigen value problem

$$S_B x_{opt} = \lambda S_W x_{opt}$$

λ is the maximal eigen value of $S_W^{-1} S_B$. The corresponding vector x_{opt} is the optimal that is beeing searched.

From this point on, all math is done as have been seen in the previous section.

5.1 Projection and reconstruction

Once M is computed the new representation s of a image r can be obtained by the next equation.

$$s_{(h,w')} = r_{(h,w)} \times M_{(w,w')}$$

Reconstruction is easily done by

$$\tilde{r}_{(h,w)} = s_{(h,w')} \times M^T_{(w,w')}$$

$\tilde{r}$ is denominated *2D fisher face*.

Figure 6 shows some reconstructed faces in the first line.

5.2 Horizontal and vertical projections

Figure 6 shows, that the feature extraction of the algorithm consists of selecting specific lines in the face. The approach mentioned by Li and Yuan is shown in the first line of this figure. Those lines are oriented vertically. By transposing all original images the algorithm would select horizontal lines. Having in mind, that information stored in the eigen vectors decreases rapidly the two projections could be used to build one classifier. The idea behind this is, instead of having one matrix which preserves 75% of variance two are used that contain each 50% of variance. In the example above this would mean: Instead of using a projection with 50% variance which would consists of 12 or 14 vectors, projection in horizantal and vertical mode of about 30% variance could be used. in this case the representation would consist of two matrices containing each 5 vectors.

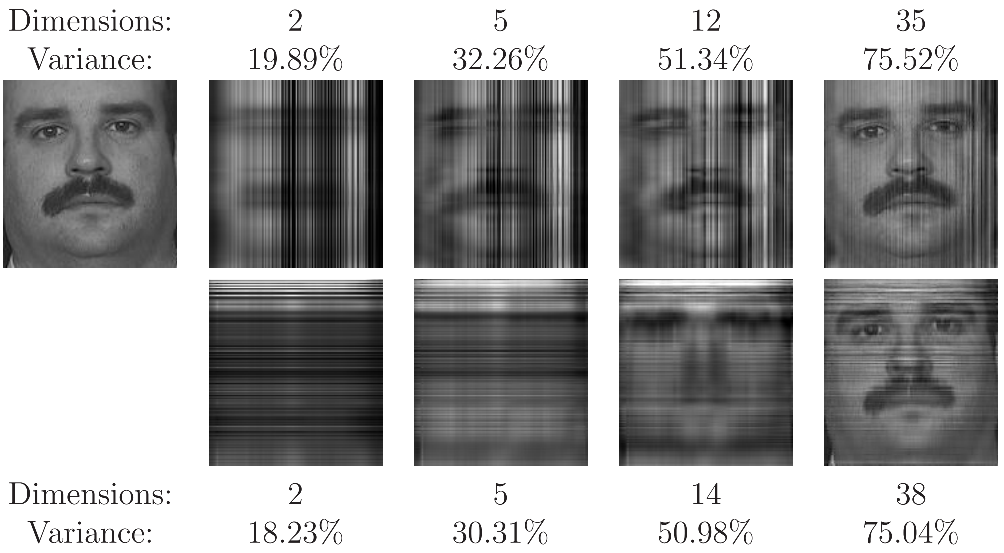

Dimensions: 2 5 12 35
Variance: 19.89% 32.26% 51.34% 75.52%
Dimensions: 2 5 14 38
Variance: 18.23% 30.31% 50.98% 75.04%

Figure 6: Sample 2D Fisherfaces

Horizontal		Vertical		Error		
Var.	Dim.	Var.	Dim.	H+V	H	V
19.89%	2	18.23%	2	3.5%	13.13%	5.88%
32.26%	5	30.31%	5	2.5%	5.25%	2.5%
51.34%	12	50.98%	14	2.25%	2.13%	2.13%
75.52%	35	75.04%	38	2.13%	2%	2.13%
90.25%	73	90.18%	74	2%	2%	1.87%

Table 1: Test results

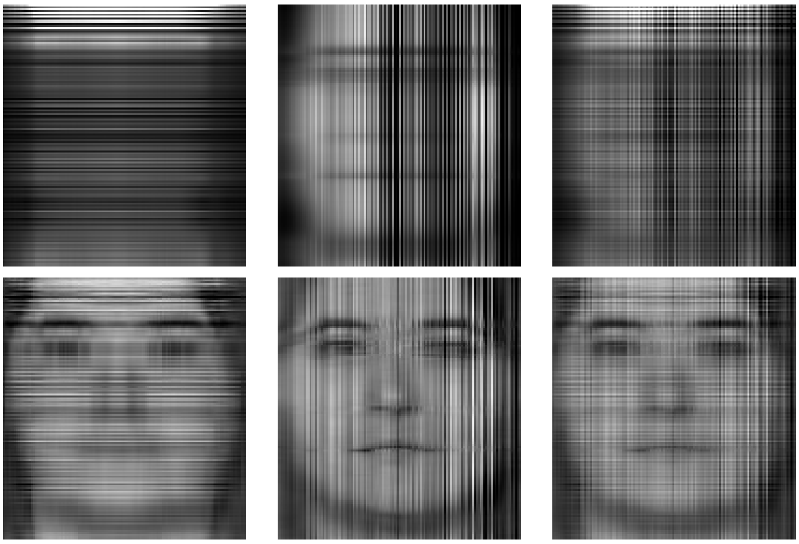
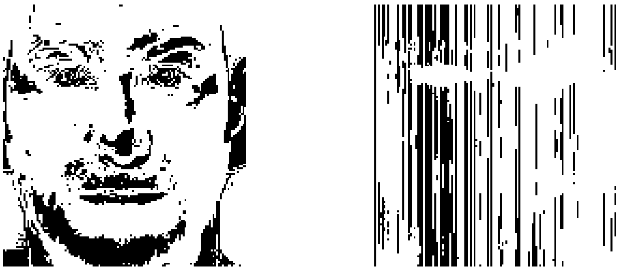


Figure 7: Horizontal, vertical and superposed projection. with variance 15% and 50%



(a) Max 25% Variance (b) 25% most accurate

Figure 8: Important regions

Table 1 shows the results of this approach. It confirms the suspection only for projections with very low variance. For variance over 30%, vertical projection outperforms horizontal and combined projection.

One possible explanation for this, is that, as the same process is being used for generating both projection matrices. Therefor both result should be similar and the features extracted should be equivalent. Figure 7 shows all three types of projections in greater detail for variance 15% and 50%.

5.3 Identifying important regions

This section tries to detect important regions that should be used for classification. While defining thouse regions can be easily done by using the variance between the original images, it is not quite that easy to detect, what regions are seen as important by the algorithm. As the classification system used in this tests is based on the nearest neighbour important regions were defined here as thouse regions that can be reconstructed with less error.

Figure 8 (a) shows the 25% pixels which most differ between all original images. Systems that can reproduce this regions should be able to perform a better classification. LDA fails to reproduce this regions as it is shown in figure 8 (b). This image was produced by marking the 25% of the pixels of the reconstructed images that had less error. As it was expected[1], large homogeneous regions like cheeks will produce less error and as a consequence they should be marked black. But for good classification, it is necessary that regions with big variance are marked also. Those regions are nearly missing at all, as can be seen in figure 8.

5.4 Symmetry

Also notably is that the information seams to be concentrated on one side. The question why one region contains discriminant information and one other very similar region (ej. cheeks) does not, arises some questions. One could think, that LDA seeks some kind of symmetry. Human faces are very symmetric. What if LDA prefers to include other regions, because the information already included is enough to reproduce the complementary region with good results? E.j. Instead of being able to project left and right cheeks with height accuracy the algorithm selects one eye and one cheek, because the other cheek and eye will be very similar to the ones stored.

If this was the case, then the distribution of the pixel with height accuracy would we ordinated in a certain way, so that the counterpart pixels, which consist of mirroring the pixel in the vertical symmetry axis, will most likely not be marked having height accuracy.

Marking 25% of the pixels in the image which are most accurate, flipping it by it vertical axis and adding both images will result in a symetric images with three types of pixels: white, symetric and asymetric ones. Non-symetric pixels are those, which are marked only in one half. Symetric pixels are marked in both sides.

On randomly created image flipped and superposed in the same way, white pixels are expected with probability of 6.25%. Table 2 shows the expected and computed values. The values were obtained by calculating the distribution using 400 images.

[1] LDA is a low-pass filter. So large homogeneouse regions will most likely be reproduced better than regions with much variation

	Expected Mean	Computed Mean
White	56.25%	57.18%
Non-Symetric	38.50%	35.63%
Symetric	6.25%	7.18%

Table 2: Symmetry check

If the above assumption would be correct it would mean, that more parts of the images are marked as non-symetric pixels, less parts marked as symetric and less pixels should be marked beeing white. Table 2 shows clearly that this is not the case.

Figure 9 also ilustrates that the probes analyzed are sufficient to computed the values in table 2. This figure shows only the distribution for white pixels of the 400 images used in this tests. It is expected, that the average percentage of white pixels should be 56.25%. In the real world this can varry. An image could have a percentage of white pixels that differ from the theoretical value. This derivation is described by the variance. Although, the computed values should behave like a *normal distributed* probe. The red line, with its average in the red vertical line (to the right) shows the approximation to the computed values. The probe fits very good to the expected *normal distribution* of the values.

The green line to the left shows what was expected for the white pixels for random images. As the probes are nearly completely outside the area delimted by the green line, the reconstructed images are not equivalent to random images also.

This means, that LDA does not operate in a *random* way, but it does not also uses symmetry in the faces to do the feature extraction.

5.5 Further pros and contras

One of the major advantages of 2DLDA is that the images are not serialised. Equations (1) and (2) show that the images are kept in its two dimensional representation. The consequences are not only that that 2D-neighbourhood is maintained but also that the algorithm allows much bigger images to be processed. It does also allow nearly unlimited number of images to be inserted into the algorithm.

This is not the case in the two approaches shown in section 3 and section 3.3. But some problems persits. There is no proof, that by summing up images to the within scater S_W, this matrix stays non-singular. In fact it could happend, that by adding one single image, the matrix becomes singular. Also this would make it impossible to invert the matrix, this problem should

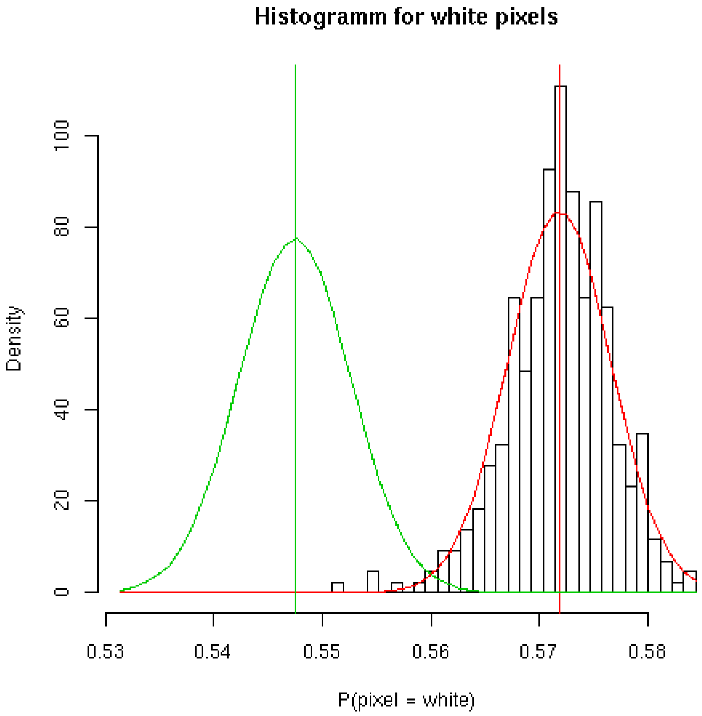


Figure 9: Distribution of occurence of white pixels

be minimised, as adding another image will most likely turn the matrix non-singular again.

A bigger problem imposes the background. If background substraction is applied, all background pixels will have the same value. A priori this is good, as they will most likely not contain discriminant information any more, so the algorithm should discard this information. But if the image has one whole line filled with background, it will turn S_W singular.

5.6 The test results

The tests results are much better than the ones shown in Li and Yuan' s original paper. The fact, that the tests done here do only consider frontal faces should be the reason for this. All test were done using 100 persons, of which four images were used for the training process and 8 for testing. All images had the size of 140 × 130 and coded in JPEG and grey scale.

Classification was done with nearest neighbour and *KNN-3*. *KNN-3* takes the three nearest neighbours. If two of them belong to a single group, this group wins. Error rates were even bigger in *KNN-3*.

References

[1] K. W. Bowyer. Face recognition technology: security versus privacy. *IEEE Technology and society magazine*, Spring:9–20, 2004.

[2] Ming Li and Baozong Yuan. Two dimesnional fisher faces. *Verbal Learning Verbal Behaviour*, 2004.

[3] Pentland A. Turk M. Eigenfaces for recognition. *Journal of Cognitive Neurosicience*, 3:71–86, 1991.

I.S.B.N. 84/689-6961-3

Facultad de Ciencias Jurídicas y Sociales
Pº Artilleros s/n
28032 Madrid
e-mail: publicaciones@fcjs.urjc.es