Wavelet Packet Analysis for Face Recognition

C. Garcia, G. Zikos, G. Tziritas

Institute of Computer Science,
Foundation for Research and Technology-Hellas (FORTH)
P.O.Box 1385, GR 711 10 Heraklion, Crete, Greece
Tel.: +30 (81) 39 17 01, Fax: +30 (81) 39 16 01

Abstract

Content-based indexing methods are of great interest for image and video retrieval in audio-visual archives, such as in the Esprit project DiVAN that we are currently developing. Detecting and recognizing human faces automatically in video data provide users with powerful tools for performing queries. In this article, a new scheme for face recognition using a wavelet packet decomposition is presented. Each face is described by a subset of band filtered images containing wavelet coefficients. These coefficients characterize the face texture and a set of simple statistical measures allows us to form compact and meaningful feature vectors. Then, an efficient and reliable probabilistic metric derived from the Bhattacharrya distance is used in order to classify the face feature vectors into person classes.

Keywords: Face recognition; Wavelet decomposition; Bhattacharrya distance

1 Introduction

Face recognition is becoming a very promising tool for automatic multimedia content analysis and for content-based indexing video retrieval system. Such a system is currently developed within the Esprit project DiVAN ([1]) which aims at building and evaluating a distributed audio-visual archives network providing a community of users with facilities to store video raw material, and access it in a coherent way, on top of high-speed wide area communication networks. The video raw data is first automatically segmented into shots and from the content-related image segments, salient features such as region shape, intensity, color, texture and motion descriptors are extracted and used for indexing and retrieving information.

In order to allow queries at a higher semantic level, some particular pictorial objects have to be detected and exploited for indexing. We focus on human faces detection and recognition, given that such data are of great interest for users queries.

In recent years, considerable progress has been made on the problem of face detection and face recognition, especially under stable conditions such as small variations in lighting, facial expression and pose. A good survey may be found in [2]. These methods can be roughly divided into two different groups: geometrical features matching and template matching. In the first case, some geometrical measures about distinctive facial features such as eyes, mouth, nose and chin are extracted ([3,4]). In the second case, the face image, represented as a two-dimensional array of intensity values, is compared to a single or several templates representing a whole face. The earliest methods for template matching are correlation-based, thus computationally very expensive and require great amount of storage and since a few years, the Principal Components Analysis (PCA) method also known as Karhunen-Loeve method, is successfully used in order to perform dimensionality reduction ([5–9]). We may cite other methods using neural network classification ([10,11]), using algebraic moments [12], using isodensity lines [13], or using a deformable model of templates ([14,15]).

In this paper, we propose a new method for recognition of frontal views of faces under roughly constant illumination. Our scheme is based on a wavelet packet decomposition of the face images. Each face image is described by a subset of band filtered images containing wavelet coefficients. From these wavelet coefficients which characterize the face texture, we form compact and meaningful feature vectors, using simple statistical measures. Then, we show how an efficient and reliable probabilistic metric derived from the Bhattacharrya distance can be used in order to classify the face feature vectors into person classes. Experimental results are presented using images from the FERET and the FACES databases. The efficiency of our approach is analyzed according to the FERET evaluation procedure and by comparing our results with those obtained using the well-known Eigenfaces method.

2 The proposed approach

In the last decade, wavelets have become very popular, and new interest is rising on this topic. The main reason is that a complete framework has been recently built ([16,17]) in particular for what concerns the construction of wavelet bases and efficient algorithms for its computation.

We based our approach on the wavelet decomposition of faces images for the reasons that we explain hereafter.

The main characteristic of wavelets (if compared to other transformations) is the possibility to provide a multiresolution analysis of the image in the form of coefficient matrices. Strong arguments for the use of multiresolution decomposition can be found in psychovisual research, which offers evidence that the human visual system processes the images in a multiscale way. Moreover, wavelets provide a spatial and a frequential decomposition of a the image at the same time.

Wavelets are also very flexible: several bases exist, and one can choose the basis which is more suitable for a given application. We think that this is still an open problem, and up to now only experimental considerations rule the choice of a wavelet form. However, the choice of an appropriate basis can be very helpful.

Computational complexity of wavelets is linear with the number (N) of computed coefficients ($O(N)$) while other transformations, also in their fast implementation, lead to $N \times log_2(N)$ complexity. Thus, wavelets are adapted also for dedicated hardware design (Discrete wavelet Transform). If the recognition task has real time computation needs, the possibility of embedding part of the process in Hardware is very interesting, like in compression tasks ([18]).

2.1 Wavelet packet decomposition of face images

The (continuous) wavelet transform of a 1-D signal $f(x)$ is defined as:

$$(W_a f)(b) = \int f(x)\, \psi_{a,b}(x)\, dx \;\; \text{with} \;\; \psi_{a,b}(x) = \frac{1}{\sqrt{a}} \psi \left(\frac{x-b}{a} \right) \tag{1}$$

The mother wavelet ψ has to satisfy the admissibility criterion to ensure that it is a localized zero-mean function. Equation (1) can be discretized by restraining a and b to a discrete lattice ($a = 2^n, b \in \mathbf{Z}$). Typically, some more constraints are imposed on ψ to ensure that the transform is non-redundant, complete and constitutes a multiresolution representation of the original signal. This leads to an efficient real-space implementation of the transform using quadrature mirror filters. The extension to the 2-D case is usually performed by applying a separable filter bank to the image. Typically, a low filter and a bandpass filter (H and G respectively) are used. The convolution with the low pass filter results in a so-called approximation image and the convolution with the bandpass filter in a specific direction results in so-called details image.

In classical wavelet decomposition, the image is split into an approximation and details images. The approximation is then split itself into a second-level approximation and details. For a n-level decomposition, the signal is decomposed in the following way:

$$A_n = \left[H_x * [H_y * A_{n-1}]_{\downarrow 2,1}\right]_{\downarrow 1,2} \tag{2}$$

$$D_{n1} = \left[H_x * [G_y * A_{n-1}]_{\downarrow 2,1}\right]_{\downarrow 1,2} \tag{3}$$

$$D_{n2} = \left[G_x * [H_y * A_{n-1}]_{\downarrow 2,1}\right]_{\downarrow 1,2} \tag{4}$$

$$D_{n3} = \left[G_x * [G_y * A_{n-1}]_{\downarrow 2,1}\right]_{\downarrow 1,2} \tag{5}$$

where $*$ denotes the convolution operator, $\downarrow 2, 1$ ($\downarrow 1, 2$) sub-sampling along the rows (columns) and $A_0 = I\,(x, y)$ is the original image. A_n is obtained by low pass filtering and is the approximation image at scale n. The details images D_{ni} are obtained by bandpass filtering in a specific direction ($i = 1, 2, 3$ for vertical, horizontal and diagonal directions respectively) and thus contain directional detail information at scale n. The original image I is thus represented by a set of subimages at several scales: $\{A_n, D_{ni}\}$.

The *wavelet packet decomposition*, that we perform in our approach, is a generalization of the classical wavelet decomposition that offers a richer signal analysis (discontinuity in higher derivatives, self-similarity,...). In that case, the details as well as the approximations can be split. This results in a wavelet decomposition tree. Figure 1 shows the H and G filters that have been applied. These filters have been selected based on trials during our experimentation.

Figure 1. H (solid line) and G (dashed line) filters

Usually, an entropy-based criterion is used to select the deepest level of the tree, while keeping the meaningful information.

In our experiments, a two levels wavelet packet decomposition is performed, as shown in figure 2. There is no need to perform a deeper decomposition because, after the second level, the size of images is becoming too small and no more valuable information is obtained. At the second level of decomposition, we obtain one image of approximation (low-resolution image) and 15 images of details which are displayed in figure 3.

Figure 2. A wavelet packet tree

Figure 3. Level 2 of the wavelet packet tree

Therefore, the face image is described by 16 wavelet coefficient matrices, which represent quite a huge amount of information (equal to the size of the input image). It is well-known that, as the complexity of a classifier grows rapidly with the number of dimensions of the pattern space, it is important to take decisions only on the most essential, so-called discriminatory information, which is conveyed by the extracted features. Thus, we are faced with the need of dimensionality reduction. Each of the 16 coefficient matrices contains information about the texture of the face. An efficient way of reducing dimensionality and characterizing textural information is to compute, for each matrix of wavelet coefficients, a set of statistical features (moments) and in order to build face features vectors as explained in the next section.

2.2 Facial features localization

Before proceeding with wavelet packet analysis and feature extraction, we aim at locating different specific areas of the face from which the wavelet coefficients will be analyzed and our measurements will be extracted. We have chosen to locate the face bounding box and then to divide this face bounding box into two areas, the top part and the bottom part, where the frontier between these two areas is the nose baseline. In order to perform this task, we search for the left and right borders of the face and for facial features, using a set of constraints like bilateral symmetry of the face, the presence of two eyes, one nose and one mouth, as well as anthropometric measures and horizontal alignment constraints for the eyes in the case of frontal views.

Lots of algorithms have been proposed to solve the problem of facial features extraction, mainly based on template matching as described in [3]. Templates are built for each of the facial parts like the eyes, the mouth and the nose. Intensity normalization is required as well as a set of different sizes templates to scope with scale variations. Then, correlation is used to locate these templates in the face image, which is a computationally expensive technique. Additionally, facial features of different people can be quite different.

For fast computation needs, our goal is to roughly locate the facial features by using simple image processing techniques. The useful technique of integral projection for the extraction of facial features has been proposed first by Kanade in [20]. For an image $I\left(x,y\right)$, the horizontal and vertical projection, respectively $H\left(y\right)$ and $V\left(x\right)$ over the image area $\mathcal{A} = [x_1, y_1] \times [x_2, y_2]$ are defined as:

$$\forall \left\{x,y\right\} \in \mathcal{A}, \quad H\left(y\right) = \sum_{x=x_1}^{x_2} I\left(x,y\right) \quad \text{and} \quad V\left(x\right) = \sum_{y=y_1}^{y_2} I\left(x,y\right) \tag{6}$$

The integral projection results are then analyzed to locate facial features. In the work of Kanade ([20]), integral projections are performed on a binary picture, obtained after applying a Laplacian operator and thresholding the result. In the work of Brunelli and Poggio ([3]), the Laplacian operator is used as well as the edge map which is then segmented according to edge directions (gradient direction). Horizontal gradients are then used to detect the left and right boundaries of the face and vertical gradients are used to detect the head top, eyes, nose and mouth baselines. Integral projections are used to check the results of a costly template matching stage.

In our approach, we take advantage of the band filtered images obtained after wavelet packet decomposition. More precisely, we perform an integral vertical projection on the vertical details image of level 1 (filtering in the horizontal direction). From the resulting projection vector $V(x)$, we search for the two extreme local maxima (peaks) corresponding to the left and the right borders of the face. These two positions define a vertical band we call *bounding band*, in which the face is contained. Then, we search for specific facial features inside this bounding band. We consider the horizontal details image of level 1 (filtering in the vertical direction) for each face image. There are high coefficients values in the area of the eyes, the nose and the mouth. We use as well the constraints of horizontal alignment of these features, in the case of frontal views (up to an inclination angle of 15^o). We perform a horizontal integral projection of the wavelet coefficients, inside the bounding band. Then, we search for local maxima in the resulting projection vector $H(y)$, using a priori knowledge about the facial features disposition. More precisely, we look for three local maxima (peaks) having a known relative vertical disposition: the baseline of the eyes, the baseline of the nose and the baseline of the mouth. Then, vertical integral projections are performed on the horizontal details image in a horizontal band around each baseline in order to check the results. In the case of the eyes, two salient peaks have to be obtained, while in the case of the nose and the mouth a large peak has to be found at the mid-point of the peaks corresponding to the two eyes positions.

In figure 4, we present the results for different faces extracted from the FERET database. For each face, the first image is the level 1 horizontal details image containing the wavelet coefficients. The central image displays the vector of horizontal projection and the right image corresponds to the three peaks we selected, overlaid on the original face image with white lines.

Figure 4. Three faces and the selected facial features baselines

Finally, we define the bounding box of the face, by closing the bounding band. It is quite difficult to precisely find the top and the bottom baselines of the face, because of the presence of hair, beard and cloths which provide a lot of high wavelet coefficients in most of the directions. To overcome this problem, we choose the top and bottom lines of the bounding box by using knowledge about human faces shape. We consider that the height of the bounding box should be approximately 1.5 times larger than its width. By this way, we place the top and bottom lines at equal distance from the nose baseline and with a distance of 1.5 times the width of the bounding box from one to the other. In figure 5, we show the different areas that we select for a given face. Then, we may divide the face in two different areas, using the nose baseline, as a frontier.

Figure 5. Bounding box, top and bottom areas

2.3 Feature vectors extraction

First, we extract 6 statistical measures from the level 2 approximation image in three different areas as shown in figure 5. From the bounding box, we extract two areas : a border area (referred as *out*) whose width is a percentage of the bounding box width, typically 15% and the interior area which is the remaining part of the bounding box area (referred as *in*) with less including hair or background. The border area of the bounding box will give information about the face shape and the interior area will provide information about the face texture and the skin-tone. The two others areas are the top part and the bottom part of the face, included in the *in* area and separated by the nose baseline. From the border area and the top and bottom areas, we extract the mean values μ_{out}, μ_{top}, μ_{bottom} and the corresponding variances σ^2_{out}, σ^2_{top}, σ^2_{bottom} of the wavelet coefficients contained in the approximation image.

From the other 15 detail images, we extract the means μ_i and variances σ_i (i=3,..,17) from the whole bounding box area. In fact, the mean values μ_i of the details images are null, due to the design of the bank filters that we apply. Thus, the feature vectors contain a maximum of 21 components (3 means and 3 variances for the approximation image and 15 variances for the details images) and are described as follows:

$$\mathcal{V} = \bigcup_{i=0}^{17} \left\{ \mu_i, \sigma_i^2 \right\} \tag{7}$$

where $\forall i \geq 3, \mu_i = 0$, and indices $i = 0, 1, 2$ stand respectively for the *out, top* and *bottom* means and variances.

In fact, after the extraction of all the vectors of the training set, we keep the most meaningful components by checking the mean value of each of them for all the feature vectors. Only the vector components with a mean value above a predefined threshold are included in the feature vector. Typically, feature vectors of size 11 are built for a threshold value of 0.9.

When solving a pattern recognition problem, the ultimate objective is to design a recognition system which will classify unknown patterns with the lowest possible probability of misrecognition. In the feature space defined by a set of features $X = [x_1, ..., x_n]$ which may belong to one of the possible m pattern classes $\omega_i, i = 1, .., m$, an error probability can be defined but can not be easily evaluated ([21]). Thus, a number of alternative feature evaluation criteria have been suggested in the literature [21]. One of these criteria is based on probabilistic distance measures.

It is easy to show that, in the two-classes case, the error probability e can be written:

$$e = \frac{1}{2}\left[1 - \int \left| p\left(X|\omega_1\right) P\left(\omega_1\right) - p\left(X|\omega_2\right) P\left(\omega_2\right) \right| \, dX \right] \tag{8}$$

According to equation (8), the error will be maximum when the integrand is zero, that is, when density functions are completely overlapping, and it will be zero when they don't overlap. The integral in (8) can be considered as the probabilistic distance between the two density functions.

In our approach, the *Bhattacharyya distance* $\mathcal{B}$ is chosen as a probabilistic distance:

$$\mathcal{B}\left(X\right) = -\ln \int \left[p\left(X|\omega_1\right) p\left(X|\omega_2\right)\right]^{\frac{1}{2}} dX \tag{9}$$

In the multi-classes case and in order to solve our problem, we make the assumption that the class-conditional probability distributions are Gaussian, that is, when the density functions are defined as:

$$p\left(X|\omega_i\right) = \left[\left(2\pi\right)^n \left|\Sigma_i\right|\right]^{-\frac{1}{2}} \times \exp\left\{{-\frac{1}{2}\left(X-\mu_i\right)^T \Sigma_i^{-1} \left(X-\mu_i\right)}\right\} \tag{10}$$

where μ_i and Σ_i are the mean vector and covariance matrix of the i^{th} class distribution respectively. The multivariate integrals in the measure can be evaluated, which leads to:

$$B = \frac{1}{4}\left(\mu_2 - \mu_1\right)^T \left[\Sigma_1 + \Sigma_2\right]^{-1} \left(\mu_2 - \mu_1\right) + \frac{1}{2}\ln\left[\frac{\left|\frac{1}{2}\left(\Sigma_1 + \Sigma_2\right)\right|}{\sqrt{|\Sigma_1||\Sigma_2|}}\right] \tag{11}$$

We consider that each component pair $\{\mu_i, \sigma_i^2\}$ is independent from the other component pairs of the feature vector $\mathcal{V}$. Thus, the distance between to feature vectors $\mathcal{V}_k$ and $\mathcal{V}_l$ is computed on a component-pair basis, that is, the distance is considered as a sum of distances relative to each of these component pairs. Using the Bhattacharrya distance, the distance $\mathcal{D}_i$ between the component pairs i of the two feature vectors $\mathcal{V}_k$ and $\mathcal{V}_l$ is:

$$\mathcal{D}_i\left(\mathcal{V}_k, \mathcal{V}_l\right) = \frac{1}{4}\frac{\left(\mu_{ik} - \mu_{il}\right)^2}{\left(\sigma_{ik}^2 + \sigma_{il}^2\right)} + \frac{1}{2}\ln\left[\frac{\frac{1}{2}\left(\sigma_{ik}^2 + \sigma_{il}^2\right)}{\sqrt{\sigma_{ik}^2\ \sigma_{il}^2}}\right] \tag{12}$$

As a consequence, the resulting distance $\mathcal{D}$ between two feature vectors $\mathcal{V}_k$ and $\mathcal{V}_l$ can be chosen as:

$$\mathcal{D}\left(\mathcal{V}_k, \mathcal{V}_l\right) = \sum_{i=1}^{n} \mathcal{D}_i\left(\mathcal{V}_k, V_l\right) \tag{13}$$

where n is the number of component-pairs in each feature vector.

3 Experimental Results

In order to test the efficiency of the algorithm presented above, we performed a series of experiments following the FERET evaluation procedure presented by P.J. Phillips et al. in [22]. We evaluate our algorithm using sets of images extracted from two databases of faces. The first set is extracted from the FACES database of the MIT Media Lab used in the Photobook project ([7]), and contains 600 images of 200 individuals (3 images per person). The second set is extracted from the FERET database. This is a collection of 310 images of 155 individuals (2 images per person). In these databases, the images that belong to the same person (same class) usually present variations in expression and illumination. In addition, they are not well-framed (variations in position) in the FERET database.

Sample images from the two sets are displayed in figures 6 and 7.

Figure 6. Sample images from the FACES database

Figure 7. Sample images from the FERET database

The FERET procedure tests evaluate the ability of the algorithm to recognize faces from a set of known individuals (referred as the *gallery*; an image of an unknown face presented to the algorithm is a *probe*, and the collection of probes is called then *probes set*).

3.1 Experiment 1

In this first experiment, we aims at studying how the size of the gallery affects the recognition performances. To do so, we iteratively test our program using fractions of the available images in the whole database. At each iteration, the gallery is increased by a set of new faces.

We extract the feature vectors of all the images in the gallery and then form the mean vectors of each class c (namely $\mathcal{V}_c^{mean}$), that is, we use an intra-class information. Then, we verify that each image k in the probe set (equal to the gallery) is classified into the correct class, looking for the minimum $\mathcal{D}\left(\mathcal{V}_k, \mathcal{V}_c^{mean}\right)$ distance, for each class c. More precisely, image k belongs to class c if $\mathcal{D}\left(\mathcal{V}_k, \mathcal{V}_c^{mean}\right)$ is minimum among all classes c. The results of these experiments are displayed in table 1 and table 2.

Table 1. Results of Experiment 1 on the FACES database

Table 2. Results of Experiment 1 on the FERET database

From these results, it can be seen that the recognition rates vary from 100.0% to 96.12%, with scores of 97.00% and 96.12% for the whole set of images in FACES and FERET respectively. These results are good if we consider the quite significant number of faces to be classified. In the FACES database, perfect classification is obtained if we use up to 120 images. Above all, these results are very similar for both databases which may mean that the proposed method is stable and tolerant to changes in facial expression as well as changes in position.

3.2 Experiment 2

This experiment was performed using the images of the FACES database. Since 3 images of each individual are available, the first two are used to build the gallery (as training data in order to compute the mean vector) and the third image is added to the probes set. Then, our algorithm runs over the probes set. The results are displayed in table 3. It can be seen that the recognition rate for the whole dataset decreases from 97.00% to 90.83%, which means that only two available images of each class seem not to be enough to estimate a good mean class vector, according to the face variations. Therefore, using the mean class vector seems to improve the classification results.

Table 3. Results of Experiment 2 on the FACES database

3.3 Experiment 3

As explained in [22], the closed-universe model that we are using (every probe has a at least one correspondent in the gallery), allows us to ask how good the algorithm is at identifying a probe image; the question is not always 'is the top match correct?' but 'is the correct answer in the top n matches?'. This let us know how many images have to be examined to obtain a desired level of performance. The performance statistics are reported on cumulative match scores respectively for the FACES (Fig. 8) and the FERET (Fig. 9) databases. The horizontal axis gives the rank, and the vertical axis is the percentage correct. The computation of the score is quite simple. Let P be the number of probes to be scored, and R_k the number of these probes in the subset that are in the top k. The fraction reported correctly is R_k/P.

In order to check the discriminatory properties of our scheme, we perform the features vector classification as in experiment 2, but without using any class information, that is, without computing the class mean vectors. More precisely, image k (in the probes set) belongs to class c if $\mathcal{D}\left(\mathcal{V}_k, \mathcal{V}_l\right)$ is minimum and image l (in the gallery, i.e., $l \neq k$) belongs to class c.

In the FACES database case, the gallery image includes one image for each person (200 images) and the probes set includes the two others images of this person (400 images). In Fig. 8, one can see that the correct answer is rank 1 for 81.9% of the probes scored, and the correct answer was in the top 10 for 97.9% of the probes scored. If we consider until rank 35, 100% correct classification is obtained. In the FERET database case, the gallery image includes one image for each person (155 images) and the probes set includes of the other image of this person (155 images). In Fig. 9, one can see that the correct answer is rank 1 for 80.5% of the probes scored, and the correct answer was in the top 10 for 94.9% of the probes scored. If we consider until rank 70, 100% correct classification is obtained.

Figure 8. Cumulative match score for the FACES database

Figure 9. Cumulative match score for the FERET database

3.4 Comparison with the Eigenfaces method and discussion

In the Eigenfaces approach, each image is treated as a high dimensional feature vector by concatenating the rows of the image together, using each pixel as a single feature. Thus, each image is considered as a sample point in a high-dimensional space. The dimension of the feature vector is usually very large, on the order of several thousands for even small image sizes (in our case, the image size is $128 \times 128 = 1024$). The Eigenfaces method which uses PCA is based on linearly projecting the image space to a lower dimensional space, and maximizing the total scatter across all classes, i.e, across all images of all classes ([6,7]). The orthonormal basis vector of this resulting low dimensional space are referred as eigenfaces and are stored. Each face to recognize is then *projected* onto each of these eigenfaces, giving each of the component of the resulting feature vector. Then, Euclidian distance is used in order to classify the features vector. In figure 10, the first 6 computed eigenfaces of the FACES and FERET databases respectively are displayed.

Figure 10. The first 6 eigenfaces of the FACES and FERET databases

We applied the Eigenfaces method on both databases. We obtain very good results on the FACES database images which is actually not surprising. Indeed, in that case, the images have been normalized (well-framed) especially for the PCA method. We obtain a result of 98.17% good classification (11 errors for 600 images) using 180 eigenfaces compared to 97.0% using our approach. But, one drawback of this method is that these eigenfaces (the number of eigenfaces has to be approximately one third of the total number of images) have to be stored, which supposes an amount of extraspace in the database. A second disadvantage is that images have to be normalized. In the FERET database case, the images are not normalized as in the FACES case, and the remaining error is 92 (i.e 70.32% good) even if more than 150 eigenfaces are used. Without any normalization needs and above all without any eigenface computation and storage, the results obtained by our approach are evidently much better that those obtained by applying PCA in the FERET database case.

Another key point of our scheme, compared to the Eigenfaces method, is the compact size of the feature vectors that represent the faces and above all, the very high matching speed that we provide. Indeed, the time required to perform the wavelet packet analysis of a test image and to extract the feature vectors is of approximately 0.05 s. on a SUN-Ultra 1 workstation, while the time for comparing a test image to the whole database (600 images) is 0.082 s. The PCA method requires quite a long time of training in order to compute the eigenfaces and the recognition process is as well expensive because it is correlation-based: the test image has to be correlated with each of the eigenfaces.

4 Conclusion

Our experiments show that a small transform of the face, including translation, small rotation and illumination changes, leave the face recognition performance relatively unaffected. For both databases, good recognition rates of 97.00% and 96.12% are obtained. Thus, the wavelet transform proved to provide an excellent image decomposition and texture description. In addition to this, very fast implementations of wavelet decomposition are available in hardware form. We show that even very simple statistical features such as means and variances provide an excellent basis for face classification, if an appropriate distance is used. The Bhattacharyya distance proved to be very efficient for this purpose. As an extension of this work, we believe that it would be interesting to directly extract the statistical features from the wavelet decomposition of the facial features such as eyes, mouth and nose, especially to tackle the probal of non-frontal views recognition. That will not increase much the size of the feature vector but we will have previously to detect more precisely the features location in order to extract the values. However, detecting precisely facial features is by itself a difficult and time consuming process so this strategy will increase the time that actually will be needed for recognition. Therefore, we will focus on a fast and efficient algorithm for precise features detection.

Acknowledgements

This work was funded in part under the DiVAN Esprit Project EP 24956.

References

[1] EP 24956. Esprit Project. Distributed audioVisual Archives Network (DiVAN).
http:\\divan.intranet.gr\info, 1997.

[2] C. L. Wilson, C. S. Barnes, R. Chellappa, S. A. Sirohey. Face Recognition Technology for Law Enforcement Applications. *NISTIR 5465, U.S. Department of Commerce* (1994).

[3] R. Brunelli, T. Poggio. Face Recognition: Features versus Templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , **15(10)** (1993) 1042–1052.

[4] J. Cox, YJ. Ghosen, P. Yianilos. Feature-based face recognition using mixture distance. *Proceedings Computer Vision and Pattern Recognition 96* (1996)

209-216.

[5] M. Kirby, L. Sirovich. Application of the Karhunen-Loeve Procedure and the Characterization of Human Faces. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , **12(1)** (1990) 103–108.

[6] M. Turk, A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Science*, **3(1)** (1991) 71–86.

[7] A. Pentland, R.W. Picard, S. Sclaroff. Photobook: Content-Based Manipulation of Image Databases. in: *Proceedings of the SPIE Storage and Retrieval and Video Databases II*, **2185** (San Jose, CA, 1994).

[8] D. L. Swets, J. Weng. Using Discriminant Eigenfeatures for Image Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , **18(8)** (1996) 831–836.

[9] P. Belhumeur, J. Hespanha, D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , **19(7)** (1997) 711–720.

[10] J.L Perry, J.M Carney. Human Face Recognition Using a Multilayer Perceptron. in: *IJCNN*, (Washington D.C., 1990) 413–416.

[11] Y. Dai, Y. Nakano. Recognition of facial images with low resolution using a Hopfield memory model. *Pattern Recognition* **31(2)**(1998) 159–167.

[12] Z.Q Hong. Algebraic feature extraction of image for recognition. *Pattern Recognition* , **24(3)** (1991) 211–219.

[13] O. Nakamura, S. Mathur, and T. Minami. Identification of human faces based on isodensity maps. *Pattern Recognition* , **24(3)** (1991) 263–272.

[14] A. Lanitis, C.J. Taylor, T. F. Cootes. Automatic Interpretation and Coding of Face Images Using Flexible Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , **19(7)** (1997) 743–756.

[15] L. Wiskott, JM. Fellous, N. Kruger, C. Von der Malsburg. Face Recognition by Elastic Bunch Graph Matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , **19(7)** (1997) 775–779.

[16] Mallat S., Multifrequencies Channel Decompositions of Images and Wavelets Models. *IEEE Transactions on Acoustics, Speech and Signal Processing,* **37(12)** (1989).

[17] I. Daubechies. The Wavelet Transform, Time-Frequency Localization and Signal Analysis. *IEEE Transactions on Information Theory* , **36(5)** (1990) 961-1005.

[18] M. Ferretti, D. Rizzo. Wavelet Transform Architectures: A system Level Review. *Proc. of the* 9^{th} *International Conference ICIAP'97*, **2**, (Florence, Italy, 1997) 77-84.

[19] A. Gersho, R.M. Gray. Vector Quantization and Signal Compression. *Kluwer Academic Publisher* (1992).

[20] T. Kanade. Picture processing by computer complex and recognition of human faces. *Tech. Rep., Kyoto Univ., Dept. Inform. Sci.* , (1937)

[21] Y. Fu. Handbook of Pattern Recognition and Image Processing. *Academic Press* (1986).

[22] P.J. Phillips, H. Wechsler, J. Huang, P.J. Raus. The FERET database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing* , **16** (1998) 295–306.

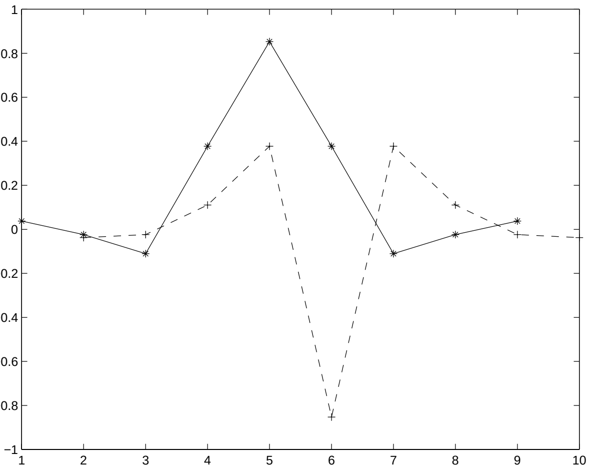


Fig. 1. H (solid line) and G (dashed line) filters

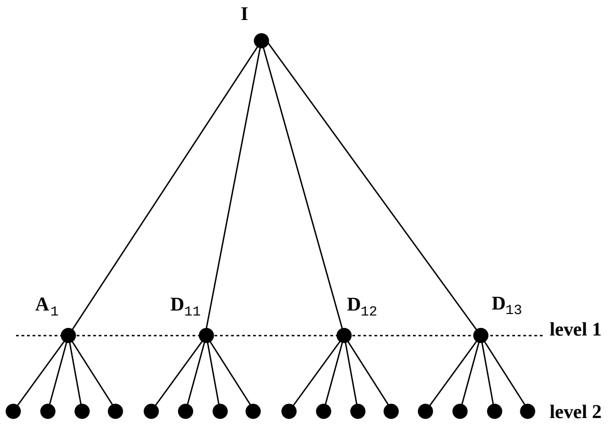


Fig. 2. A wavelet packet tree

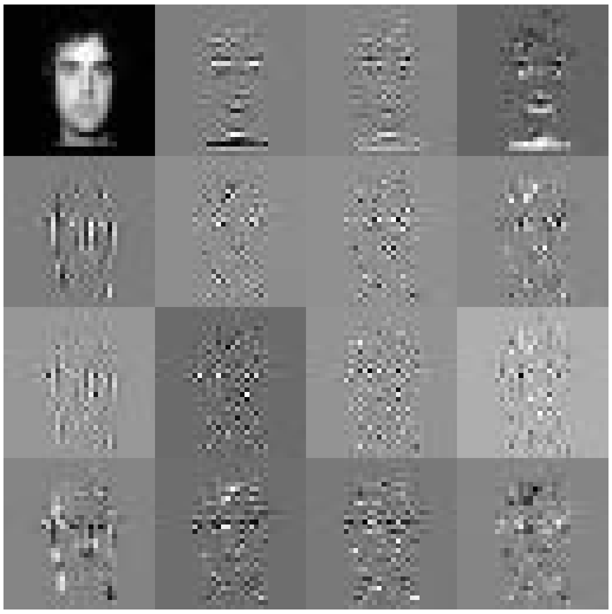


Fig. 3. Level 2 of the wavelet packet tree

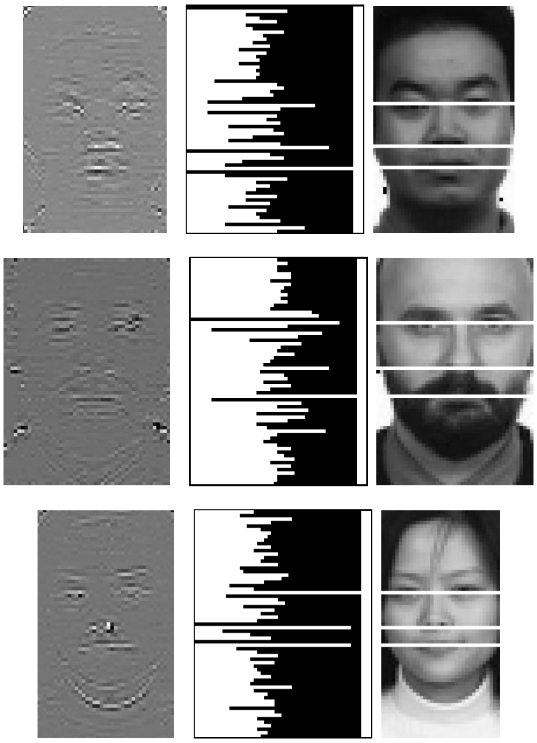


Fig. 4. Three faces and the selected facial features baselines

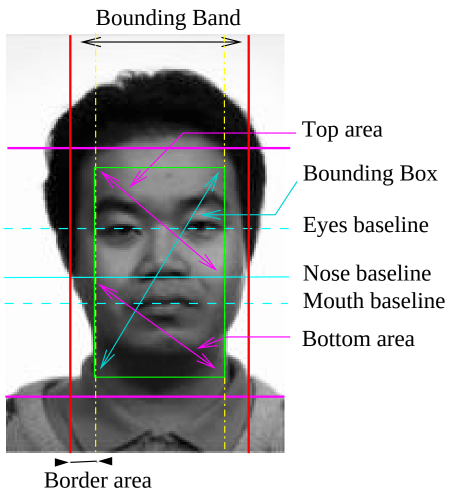


Fig. 5. Bounding box, top and bottom areas

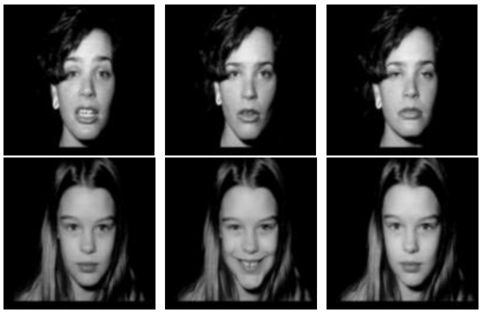


Fig. 6. Sample images from the FACES database

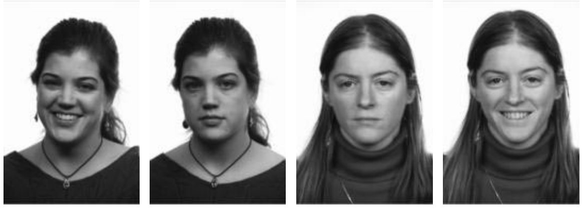


Fig. 7. Sample images from the FERET database

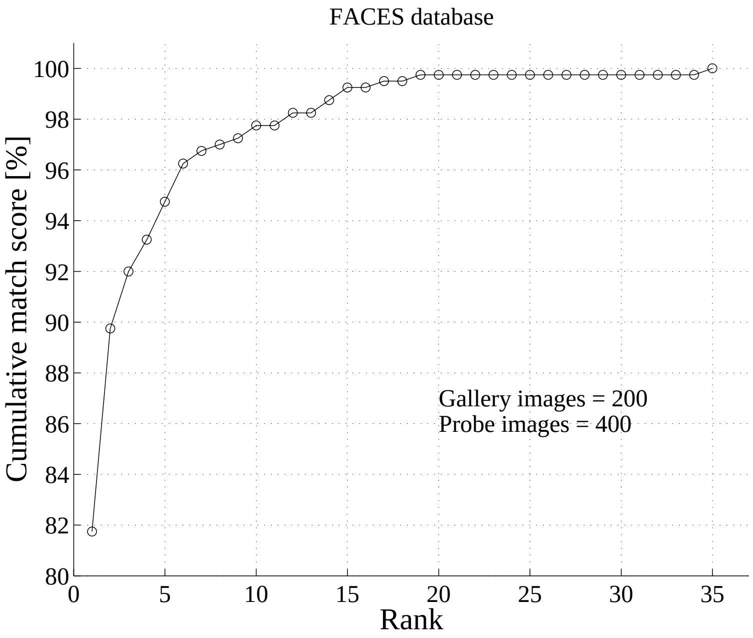


Fig. 8. Cumulative match score for the FACES database

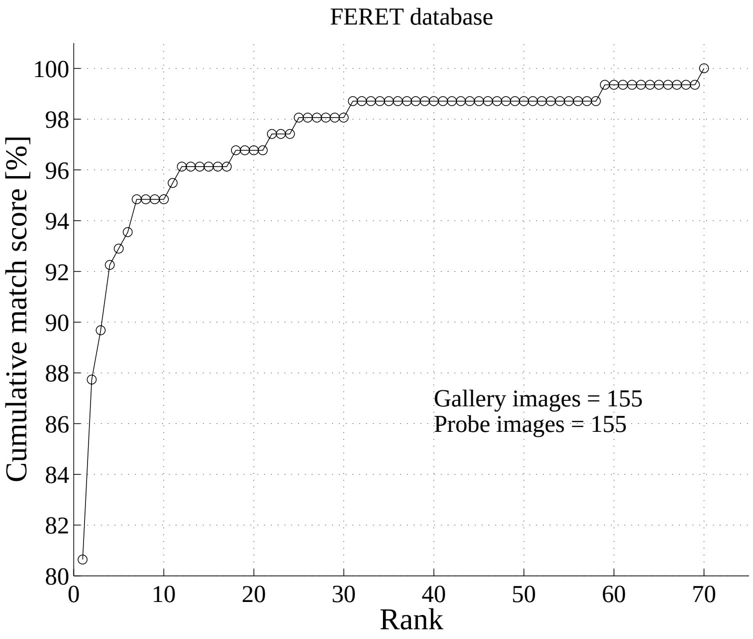


Fig. 9. Cumulative match score for the FERET database

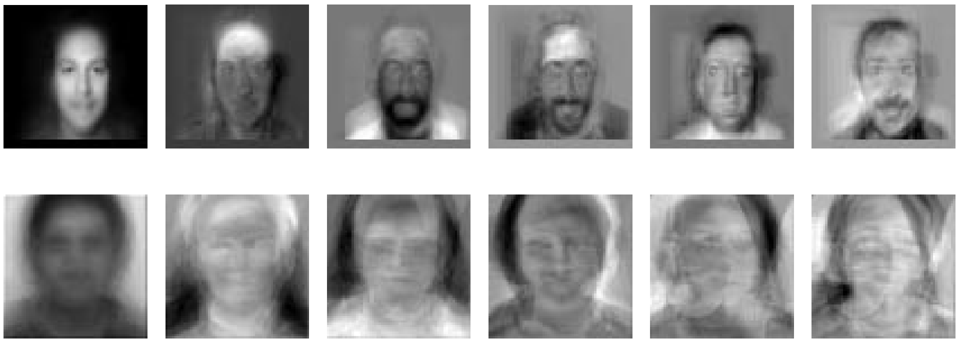


Fig. 10. The first 6 eigenfaces of the FACES database (first line) and the FERET database (second line)

Table 1
Results of Experiment 1 on the FACES database

A = Number of images, B = False alarm, C = Recognition rate

A	B	C
60	0	100.00%
90	0	100.00%
150	2	98.66%
180	3	98.33%
240	4	98.33%
270	7	97.40%
330	9	97.27%
360	10	97.22%
420	10	97.61%
450	11	97.55%
480	12	97.50%
510	15	97.05%
540	15	97.22%
570	15	97.36%
600	18	97.00%

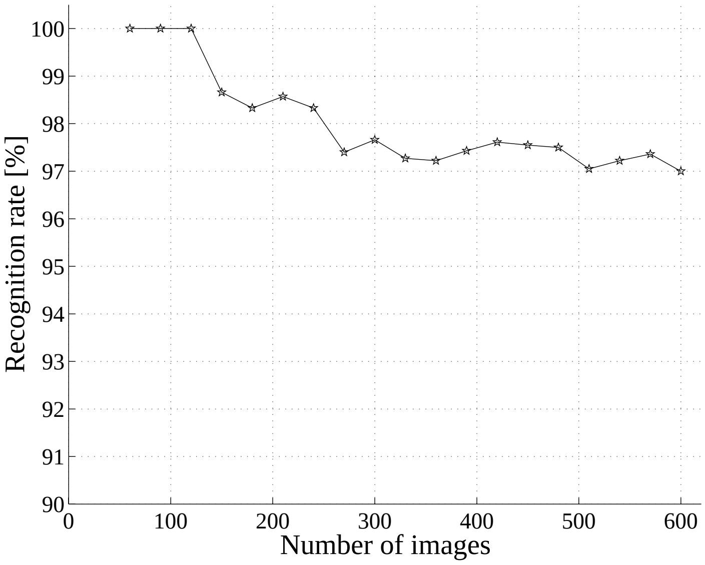


Table 2
Results of Experiment 1 on the FERET database

A = Number of images, B = False alarm, C = Recognition rate

A	B	C
60	0	100%
80	0	100%
100	0	100%
120	1	99.16%
140	1	99.28%
160	1	99.37%
180	1	99.44%
200	8	96.00%
220	8	96.36%
240	10	95.83%
260	10	96.15%
280	11	96.07%
300	12	96.00%
310	12	96.12%

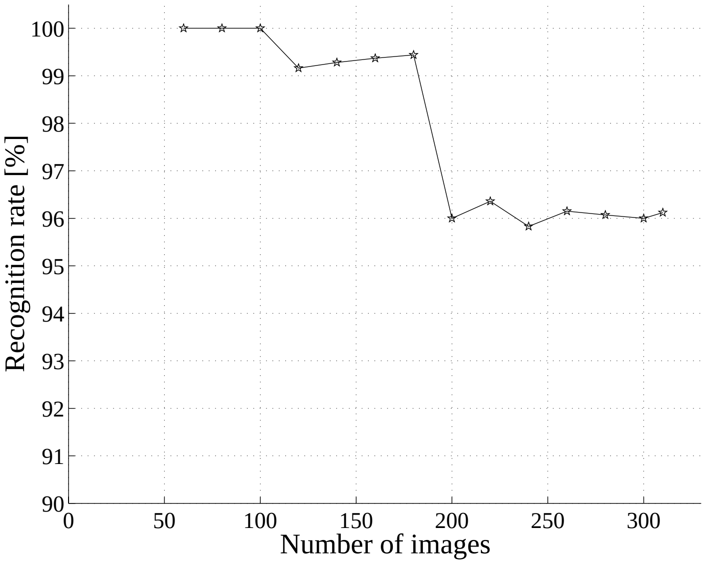
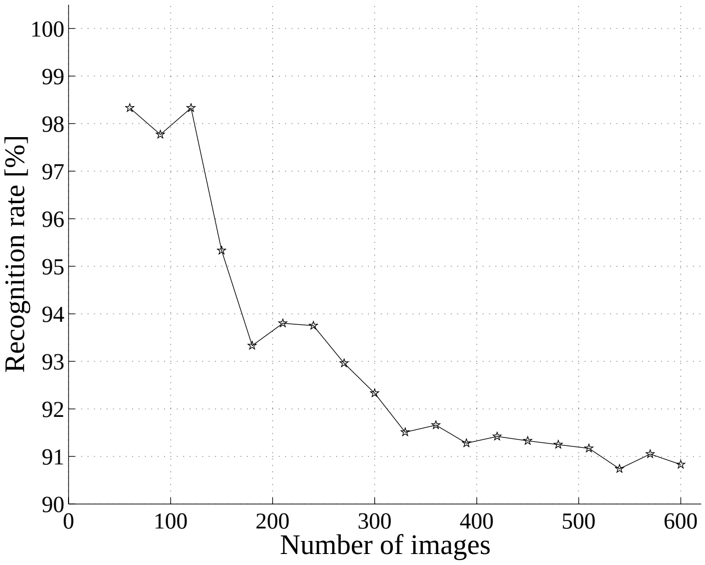


Table 3
Results of Experiment 2 on the FACES database

A = Number of images, B = False alarm, C = Recognition rate

A	B	C
60	1	98.33%
90	2	97.77%
150	7	95.33%
180	12	93.33%
240	15	93.75%
270	19	92.96%
330	28	91.51%
360	30	91.66%
420	36	91.42%
450	39	91.33%
480	42	91.25%
510	45	91.17%
540	50	90.74%
570	51	91.05%
600	55	90.83%


Recognition rate [%]
Number of images