Evaluation of the Modelling of Local Areas and Errors of Localization in FRGC'05

Onur C. Hamsici and Aleix M. Martínez
Depart. of Electrical and Computer Engineering
The Ohio State University
Columbus, OH 43210

Abstract

We present an evaluation of a probabilistic, part-based algorithm designed at The Ohio State University. Our algorithm is robust to errors of precision made by the (automatic) face and facial feature detector and to local image changes due to, for example, expression and illumination. Our contributions include the design of a novel face and facial feature detector and the justification of the use of the Mahalanobis cosine distance. We show results on experiments 1 and 4 in the FRGC (Version 2) test/database. Our algorithm includes a new face detector that is used to demonstrate the robustness of our algorithm to small errors of localization.

1 Introduction

A primary goal in the design of computer vision systems able to classify faces into a large set of classes is to automatically extract those features that vary among people yet are constant within each individual. This is, however, a task much more difficult than one may think. For example, our perception of a person's face changes as she speaks and as she expresses emotions or when faces cannot be localized with precision. And, obviously, illumination changes and pose variations also influence the way a face is seen when projected onto the CCD of a camera.

Earlier work has mainly focused on finding the subset that represents most of the faces of each individual under all possible views and all possible lighting conditions [1, 6, 18]. Recently, progress has also been made toward solving the other two problems described above; those caused by errors of localization and facial expression changes [8, 9]. In our previous work, we have tested our approach using several relatively small databases. In this paper, we will test how our method works with a large number of people and images.

In Section 2, we derive our face and facial feature detector. Section 3 summarizes our recognition approach as it applies to the problem posed by the *2005 Face Recognition Grand Challenge*. Section 4 presents the experimental results with version 1 and 2 of the database. Conclusions are in Section 5.

2 Localization of Faces and Facial features

The preprocessing step for any appearance-based face recognition algorithm requires knowledge of the position of the facial features to be used in the alignment of each face into a standard form. Our normalization procedure applies the same type of alignment as that proposed in BEE (Biometrics Experimentation Environment) in its Version 2. This calls for all eye location to be normalized to a standard position. In this section, we define a system that automatically extracts the location of the face and eyes which is to be used for such a normalization stage.

The detection of facial features, such as eyes, has proven to be a very challenging task. Furthermore, most algorithms are associated to a high computational cost and produce a large number of false detections. To solve these problems, one may use the known fact that the eyes are located on the upper half of the face. For this reason, we first derive a method to detect faces (with a false detection rate of less than 10^{-4} and a false rejection of less than 10^{-2} – for the target images in the second version of the FRGC set), and then used the top half of such locations for the detection of the eyes. To accomplish this, we define a hierarchical facial feature localization algorithm based on the idea of Subclass Discriminant Analysis (SDA).

2.1 Face Detection

To train our face detection algorithm, we randomly selected $1,024$ images from the training set in the FRGC version 2 database. Each of these faces is then rotated until both eyes are aligned with the horizontal axis of the image and scaled to have a standard intra-eye distance. The resulting set of images is then augmented by including ten new images with random (in-plane) rotations and scale changes [15]. These in-plane rotations are taken from 0^o to $\pm 15^o$, and the scale changes between 0.8 and 1.1 (similar to the method of [17]). Finally, all faces are re-scaled to a standard size of 24×24 pixels.

The process just described generates a total of $10,240$ positive training samples. We also collected a set of $10,240$ negative training samples from images of different sorts that do not contain any face. These two image sets will contain a diverse set of images of faces and non-faces that cannot be represented using a simple unimodal Gaussian distribution [15, 17]. One way to solve this, is to divide the samples in each class into a set of subclasses. We use the K-means procedure to cluster the data in each of these (positive and negative) sets into S subclasses. The idea of Subclass Discriminant Analysis (SDA) is to use cross-validation to find that value of S that optimizes classification rather than that which best approximates the true underlying distribution of the data [19]. Once the optimal value of S is determined, the basis vectors of SDA $\mathbf{V}$ can be readily obtained as

$$\Sigma_B \mathbf{V} = \Sigma_X \mathbf{V} \Lambda, \tag{1}$$

where Σ_X is the sample covariance matrix,

$$\Sigma_B = \sum_{i=1}^{S-1} \sum_{j=i+1}^{S} (\mathbf{m}_i - \mathbf{m}_j)(\mathbf{m}_i - \mathbf{m}_j)^T,$$

$\mathbf{m}_i$ is the mean feature vector of subclass i and, for simplicity, we have assumed equal priors.

The optimal number of subclasses according to SDA was 25 for the positive samples and 20 for negative ones. The first two rows in Fig. 1, show twelve of these means found by SDA (six representing the positive samples and six for the negative ones).

Note that this method will generate a single subspace. Assuming the data is linearly separable, SDA will find the optimum solution if one considers the $S - 1$ dimensions in $\mathbf{V}$. Note, however, that like LDA (Linear Discriminant Analysis) the nearest-mean classifier is only optimal when all class pdf are equal. Since this is not necessarily the case, it is convenient to employ the Mahalanobis distance instead. This means that for a new testing window $\mathbf{w}$, where $\mathbf{w}$ is an image patch of 24×24 pixels, we calculate

$$Mh(\mathbf{w}, \mathbf{m}_i) = (\mathbf{w} - \mathbf{m}_i)^T \, \widehat{\Sigma}_i^{-1} \, (\mathbf{w} - \mathbf{m}_i) \, ,$$

where $\widehat{\Sigma}_i = \mathbf{V}^T \Sigma_i \mathbf{V}$ and Σ_i is the sample covariance matrix of the vectors in subclass i.

Latter in the paper, we will use this method to detect the faces in the query set of Experiment 1 of the FRGC –version 2– database. These are the images that are used for recognition (or verification). In this set of $16,028$ images, our algorithm (described in this Section) was able to successfully find all faces except 133 of them while only selecting two non-face windows as faces. This gives a false rejection rate of 0.0083 and a false acceptance of $\approx\ 0.0001$. To obtain these results we used a three-level image pyramid, which consisted in resizing the original images by $1/16$, $1/20$ and $1/24$. As it is well-known by the community, this process ensures we find faces at different scales.

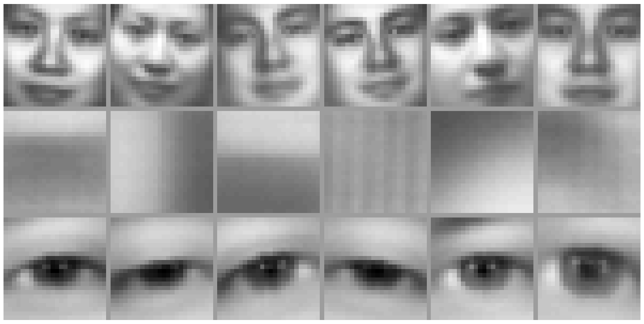


Figure 1: Some of the cluster means found by the K-means clustering step in SDA. Face cluster means are shown in the first row, background ones in row 2, and eye's in row 3.

2.2 Facial Feature Detection

After all the faces have been located using the algorithm just described, we are in a position to define a method that automatically detects the position of each of the two eyes in each of the images. Similar to what we did above, we used $1,024$ positive training images and the same set of $10,240$ negative samples defined earlier. In this case, the positive samples were drawn from those images with most uniform lighting (the so called "passport photos" in FRGC). As we did above, we also generate five new sample images of each eye by means of random in-plane rotations (0^o to $\pm 15^o$) and scales (0.8 and 1.1 times the original size). This gives us a total of $10,240$, 24×24-pixels, positive training samples for both eyes.

Note that since we used SDA, we do not need to specifically design a right and a left eye detector. These will be automatically generated by the clustering procedure in SDA. The SDA algorithm automatically divided the positive and negative samples into 25 subclasses each. The bottom row in Fig. 1 shows six of the means of these eye-subclasses.

We tested our eye detector on the face patches automatically detected by the procedure defined above. Out of a total of $16,028$ query images (of Experiment 1), our algorithm only missed 226 eyes. The number of false detection was 0. This gives a false detection rate of 0 and a false rejection rate of $\approx\ 0.0141$. Note that the false rejections are known, because our algorithm cannot detect the position of both eyes in such images. In this cases, we use the average location of the eyes located on the images of the training set as estimates.

3 Modelling of Face Localization Error and Local Parts

Before we use any of the images for training or testing our classifier, it is convenient to normalize each of the images with respect to its mean and variance. More formally, let $\mathbf{I}_{i,j}$ be the j^{th} image in class i in its vector form; i.e., $\mathbf{I}_{i,j} \in \Re^p$, where p is the number of pixels in the image. We can then obtain the normalized feature vector as

$$\mathbf{x}_{i,j,r} = \frac{\bar{\mathbf{x}}_{i,j,r}}{\sqrt{\frac{1}{p-1}\sum_{l=1}^{p} \bar{\mathbf{x}}_{i,j,l}^2}}, \tag{2}$$

where

$$\bar{\mathbf{x}}_{i,j,r} = \mathbf{I}_{i,j,r} - \frac{1}{p}\sum_{l=1}^{p} \mathbf{I}_{i,j,l}, \tag{3}$$

and $\mathbf{I}_{i,j,r}$ is the r^{th} feature of $\mathbf{I}_{i,j}$.

We can now define a way to estimate the subset of localization errors and a method to calculate similarity between each of these subsets and a new (test) feature vector.

3.1 Errors of localization

All face localization algorithms have an associated error of precision; meaning that they cannot localize every single facial feature with pixel precision. Unfortunately, test feature vectors generated from imprecisely localized faces can be closer to the training vector of an incorrect class. This problem is depicted in Fig. 2. In this figure, we display two classes, one drawn using crosses and the other with pentagons. For each class, there are two learned feature vectors, each corresponding to the same image but accounting for different errors of localization. The test image (which belongs to the "cross" class) is shown as a square. Note that while one of the "pentagon" samples is far from the test feature vector, the other corresponds to the closest sample; that is, while one localization leads to a correct classification, the other does not. This point becomes critical when the learning and testing images differ on facial expression or illumination as well as for duplicates (images of an individual taken at a latter time), which are the typical variations present in the FRGC set.

In order to tackle this problem, some authors have proposed to filter the image before doing recognition. Filtered images are somewhat less sensitive to the localization problem, but do not solve the problem completely. In [8], we proposed to model the subset of the localization problem described above by means of a probability density function.

To present our method formally, let us first assume that the localization errors for both the x and y image axes are known. We will denote these errors as vr_x and vr_y. These two parameters can be interpreted as follows. We know that, statistically, the correctly localized face falls within the set of images $\hat{\mathbf{x}}_{i,j} = \{\mathbf{x}_{i,j,1}, \ldots, \mathbf{x}_{i,j,r}\}$, where i specifies the class, $i = \{1, \ldots, c\}$, c represents the number of classes, j the j^{th} sample in class i, and r the number of possible localizations when varying the localization error from $-vr_x$ to $+vr_x$ about the x image axis and from $-vr_y$ to $+vr_y$ about the y axis.

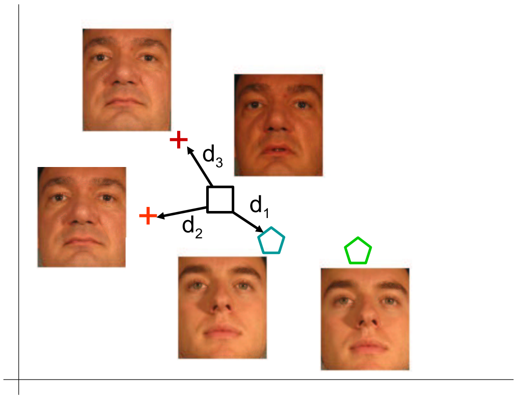


Figure 2: Different localization results lead to different representations onto the feature space. While some result in correct classification, others do not.

Once this data set, $\hat{\mathbf{x}}_{i,j}$, has been generated, the subset where the j^{th} sample of class i lies can be readily modelled by means of a Gaussian distribution whose sample mean and sample covariance matrix are

$$\begin{aligned} \mu_{i,j} &= \frac{\sum_{l=1}^{r} \mathbf{x}_{i,j,l}}{r}, \\ \Sigma_{i,j} &= \frac{1}{r-1}(\hat{\mathbf{x}}_{i,j} - \mu_{i,j})(\hat{\mathbf{x}}_{i,j} - \mu_{i,j})^T. \end{aligned} \tag{4}$$

Recall, we still need to determine the error of our localization algorithm. Obviously, this is a difficult task that unfortunately does not have an analytical solution for most face localization algorithms. Nonetheless, if the correct localization values are known (i.e., the ground-truth is known) for a set of s samples, $\mathbf{x}(s) = \{\mathbf{x}_1, \ldots, \mathbf{x}_s\}$, an estimation $E(r; \mathbf{x}(s))$ can be computed, which depends on the number of samples s. It is easy to show that $E(r; \mathbf{x}(s))$ approximates the true value of r and those of vr_x and vr_y when $s \to \infty$. Obviously, we do not have that much data and, therefore, only estimations can be made. The localization algorithm described above was found to have an associated error of 5.62 and 5.45 for the left and right eye resp. Note that this error can be interpreted as the radius of a circle centered in the middle of each of the eyes. The standard deviation of these errors were 16.66 and 13.77, and the median of the errors were 4.15 and 4.02 resp.

3.2 Subspace for image representation

If we want to use an appearance-based approach for recognition (where each image pixel represents a feature of the original feature space), it is convenient to first use a feature extraction algorithm (i.e., a subspace method) to reduce the dimensionality of such image representation. Since our n training samples will at most span a space of $n-1$ dimensions, it is reasonable to reduce the original space $\Re^p$ to $\Re^q$ where $q < n$ and generally $n << p$. In this paper we will use the subspace generated by Principal Component Analysis (PCA) [5, 14, 16]. We have previously shown that PCA is as effective as LDA and ICA when used within the probabilistic framework defined above [10].

Let Φ_{PCA} denote the projection matrix obtained using the PCA approach. Φ_{PCA} can be readily computed from the sample covariance matrix, which is given by

$$\Sigma_X = \sum_{i=1}^{c} \sum_{j=1}^{m_i} \sum_{g=1}^{G} \Sigma_{i,j,g} \, , \qquad (5)$$

where m_i is the number of samples in class i. Since the rank of these covariance matrices is typically less than the number of dimensions, it is impossible to compute the principal components directly from them. Therefore, we generally calculate the eigenvectors of $\mathbf{X}^T\mathbf{X}$ instead; where $\mathbf{X} = \{\hat{\mathbf{x}}_{1,1}, \ldots, \hat{\mathbf{x}}_{1,m_1}, \ldots, \hat{\mathbf{x}}_{c,m_c}\}$. In such cases, if λ_h and $\mathbf{e}_h$ are the h^{th} eigenvalue and eigenvector of $\mathbf{X}^T\mathbf{X}$, λ_h and $\lambda_h^{-1/2}\mathbf{X}\mathbf{e}_h$ are the eigenvalues and eigenvectors of Eq. (5) [12].

3.3 Local Parts

It is also known that facial expressions affect some areas of the face more than others [3, 4]. We can take advantage of this and try to use those areas of the face that change the least [9]. While it is generally difficult (or computationally expensive) to know which areas have changed the most in each image shot, one can use a part-based configuration to address this problem. In this approach, we divide the face into a set of local areas. Then, compute the probability of each local area to belong to each class. And, finally, combine all this information using our probabilistic framework.

In this framework, we will first divide each face image into K local parts and, then, apply the method defined above. This will require the estimation of the K subsets, and corresponding weighted subspaces, which account for the localization error and facial expression changes. Let us define the set of images of each local area as

$$\mathbf{X}_{i,j,k} = \{\mathbf{x}_{i,j,k,1}, \ldots, \mathbf{x}_{i,j,k,r}\},$$

where $\mathbf{x}_{i,j,k,l}$ is the k^{th} local area of the j^{th} sample of class i.

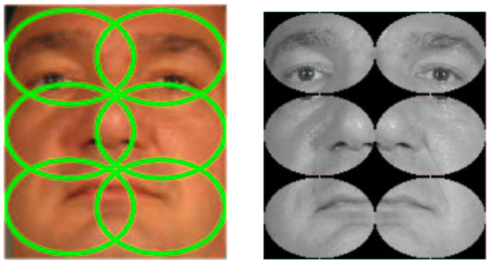

(a) (b)

Figure 3: Shown here are the six local parts used in our experiments for one of the images. (a) Position of the local areas. (b) Cropped areas; i.e., $\mathbf{x}_{i,j,k}$, $\forall k$.

This is done after we have localized and warped all face images to a standard size. To obtain each of the samples $\mathbf{x}_{i,j,k,m}$, ellipse-shaped segments as defined by $x^2/d_x^2 + y^2/d_y^2 = 1$ are used, Fig. 3; where d_x and d_y are the free parameters that define the ratio of an ellipsoid. These segments are then represented in vector form. The new mean feature vectors and covariance matrices are given by

$$\begin{aligned}\mu_{i,j,k} &= \frac{1}{r}\sum_{m=1}^{r} \mathbf{x}_{i,j,k,m} \, , \\ \Sigma_{i,k} &= \frac{1}{r-1}(\mathbf{X}_{i,j,k} - \mu_{i,j,k})(\mathbf{X}_{i,j,k} - \mu_{i,j,k})^T .\end{aligned}$$

To compute the subspaces, we will need to calculate a total of K projection matrices Φ_{PCA_k}, $k = \{1, \ldots, K\}$. In such cases, and as described in the previous section, we will need to project the estimates of our subsets onto each of the above computed subspaces,

$$\mathcal{G}_{i,j,k} = \{\widetilde{\Sigma}_{i,j,k}, \widetilde{\mu}_{i,j,k}\},$$

where $\widetilde{\Sigma}_{i,j,k} = \Phi_{a_k}^T \Sigma_{i,j,k} \Phi_{a_k}$, and $\widetilde{\mu}_{i,j,k} = \Phi_{a_k}^T \mu_{i,j,k}$. The set $\mathbf{X}_{i,j,k}$, which is generally very large, does not need to be stored in memory. $\mathcal{G}_{i,k} = \{\mathcal{G}_{i,1,k}, \ldots, \mathcal{G}_{i,m_i,k}\}$ is the subset of all images of the k^{th} local area of class i under all possible errors of localization.

3.4 Class-sample similarity

When a test face image $\mathbf{t}$ is to be classified, we project each of the local areas of $\mathbf{t}$ (i.e., $\mathbf{t}_k$, $k = \{1, \ldots, K\}$) onto their corresponding subspace

$$\tilde{\mathbf{t}}_k = \Phi_k^T \mathbf{t}_k \, .$$

Since our feature vectors $\{\mathbf{x}_{i,j}\}_{\forall i,j}$ have been normalized to lay on the surface of a $(p-2)$-dimensional sphere $\mathbf{S}^{p-2}$, it is convenient to use the Mahalanobis cosine distance defined as

$$MhCos\left(\hat{\mathbf{t}}_k, \hat{\mu}_{i,j,k}\right) = \frac{\hat{\mathbf{t}}_k^T \hat{\mu}_{\mathbf{i},\mathbf{j},\mathbf{k}}}{\|\hat{\mathbf{t}}_k\| \|\hat{\mu}_{i,j,k}\|}, \qquad (6)$$

where

$$\hat{\mathbf{t}}_{k,l} = \frac{\tilde{\mathbf{t}}_{k,l}}{\sqrt{\lambda_l}}, \quad \hat{\mu}_{i,j,k,l} = \frac{\widetilde{\mu}_{i,j,k,l}}{\sqrt{\lambda_l}}, \qquad (7)$$

$\tilde{\mathbf{t}}_{k,l}$ is the l^{th} feature in the k^{th} local area of $\mathbf{t}$, and $\widetilde{\mu}_{i,j,k,l}$ is the l^{th} feature of the mean of the localization errors of the k^{th} local area of sample j in class i.

To see why the normalization step projects the data onto the surface of $\mathbf{S}^{p-2}$, we need to understand the role of Eqs. (2 - 3). The mean normalization step defined in Eq. (3), can be rewritten as

$$\mathbf{I}_{i,j} - \frac{1}{p}\mathbf{1}_p\mathbf{I}_{i,j} = \left(\mathcal{I}_p - \frac{1}{p}\mathbf{1}_p\right)\mathbf{I}_{i,j},$$

where $\mathcal{I}_p$ is the $(p \times p)$ identity matrix and $\mathbf{1}_p$ is a p by p matrix with all elements equal to one. Note that the matrix $\left(\mathcal{I}_p - \frac{1}{p}\mathbf{1}_p\right)$ maps the data to the null space of the vector $(1, \ldots, 1)^T$ and, hence, the dimensionality of the original space is reduced to $p - 1$ dimensions.

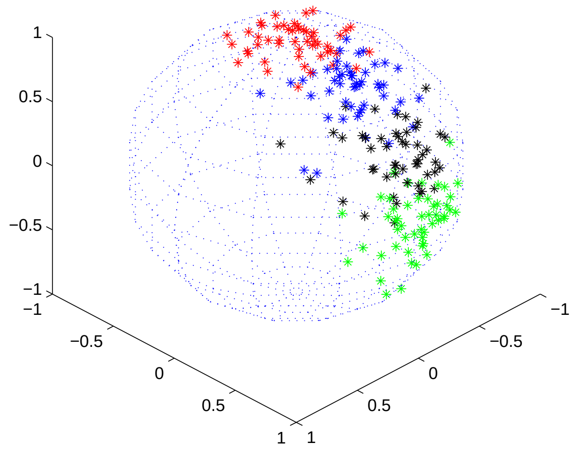
Following the mean normalization step, the variance normalization step given in Eq. (2) enforces all vectors to be at a common distances from the origin. This means that the data now lays on the surface of $\mathbf{S}^{p-2}$. Note that the surface of a sphere is always one dimension less than that of the original space.

An illustration of how some 4-dimensional vectors (belonging to four distinct classes) get mapped to such a representation is shown in Fig. 4.

Once our data has been mapped to the surface of a sphere, one should be cautious when using the PCA technique described above. Note that such techniques have been defined for hypersurfaces not hyperspheres. For example, the counter part of the Gaussian pdf in the sphere is the Kent distribution [7]. The Kent distribution is defined to represent the probability function of the density within the (localized) small ellipse circumscribing our data, which lays on the surface of a sphere. This distribution uses the mean of the points as the first *mode* of the density, defining a vector that connects the origin of $\mathbf{S}^{p-2}$ to the center of the small ellipse containing the data. Additional parameters of the density will define the variance of the data on the surface of $\mathbf{S}^{p-2}$ (i.e., the ratio defining the ellipse).

It is important to note that if we use the PCA approach defined above (which uses the sample covariance matrix) we will eliminate the mode of the density defining the mean in the Kent pdf. This is the reason why we need to be extra careful when using PCA on spherically distributed data. We could argue that one should use a better dimensionality reduction technique for spherical data. One candidate is the subspace spanned by the eigenvectors of the autocorrelation matrix. Since this matrix does not subtract the mean vector from the samples, it can preserve the mode of the density. Note that the singular vectors of an $n \times p$ data matrix obtained by Singular Value Decomposition (SVD) is the same as the eigenvectors of the autocorrelation matrix of the data; where n is the number of samples. This is the rational behind the use of the SVD approach in the computation of the illumination cone in [1]. The illumination cone takes advantage of the fact that variations along each of the dimensions of the Kent distribution on the small ellipse will represent images corresponding to variations in the (3D) location of the light source. Furthermore, the intensity of the image is already accounted for by the spherical representation, because all vectors $\alpha\mathbf{v}$ (for any $\alpha > 0$) are represented by $\mathbf{v}$ in our spherical representation defined above.

This discussion justifies the use of the cosine distance in $\mathbf{S}^{p-2}$. Note that the cosine of the angle is large for small angles (that is to say, for vectors that are alike in $\mathbf{S}^{p-2}$). However, this value of the distance will degrade rapidly after we leave the cone defined by the small ellipse circumscribing the data of the Kent distribution. Nonetheless, this means that we still need to specify the size of the small ellipse. This is given by the variance of the Kent distribution, which is the same as that of the Gaussian pdf. This was formally stated in Eq. (6) – the so-called Mahalanobis cosine distance. Note that this also justifies the empirical finding of the superiority of this distance in several experiments in



(a)

Figure 4: Shown here are the data vectors of four different classes after these have been mapped onto the surface of $\mathbf{S}^{p-2}$ using Eqs. (2 - 3).

face recognition [2, 13].

Recall that we still need to justify the use of PCA. To see this, note that when the data under study is (more or less) distributed along all directions on the surface of $\mathbf{S}^{p-2}$, the mean of the data will be (close to) the origin. In this case, the PCs of the sample covariance matrix will be (similar) to those of the autocorrelation matrix.

Finally, whitening the data before calculating the cosine distance will generally improve our recognition results, because the sample covariance matrix is the sum of the average of all within class covariance matrices and the covariance of the class means. Once we scale our space according to the eigenvalues of sample covariance matrix, the Euclidean distance in the whitened space will be a good approximate of the Mahalanobis distance in the original space (being equal in the homoscedastic case). This is the reason for the normalizing term in Eq. (7).

4 Experimental Results

A major goal is to demonstrate that: *a)* the results obtained using the probabilistic, part-based framework defined above are superior to those obtained with a global PCA method, and *b)* the results of our method do not degrade much when we compare the ROC curves obtained with manual and automatic localizations of facial features.

4.1 Experiment 1

In this first test the images used for training and testing were both captured indoors with controlled lighting [11]. This implies that we will be mainly testing the problems caused by errors of localization and those possed by different facial expressions.

The images in this first experiment are further subdivided into three groups. In the first one, query and target images were taken within the same semester. The second group contains images taken within and between semesters. And, in the third group, the target and query images were captured in different semesters. Fig. 5 shows the ROC curves for each of these three groups. In this figure we compare the results obtained with the baseline PCA approach (given by the FRGC BEE) and the method described in this paper (OSU). Note that our algorithm produces two distinct curves – that obtained with manual detection of the eyes and that generated with automatic detection. We see that our probabilistic part-based framework does indeed improve the performance of the PCA algorithm. We also note that our method is still superior when using the automatic detector defined in Section 2. This is relevant because the results of the baseline where obtained using manual detections.

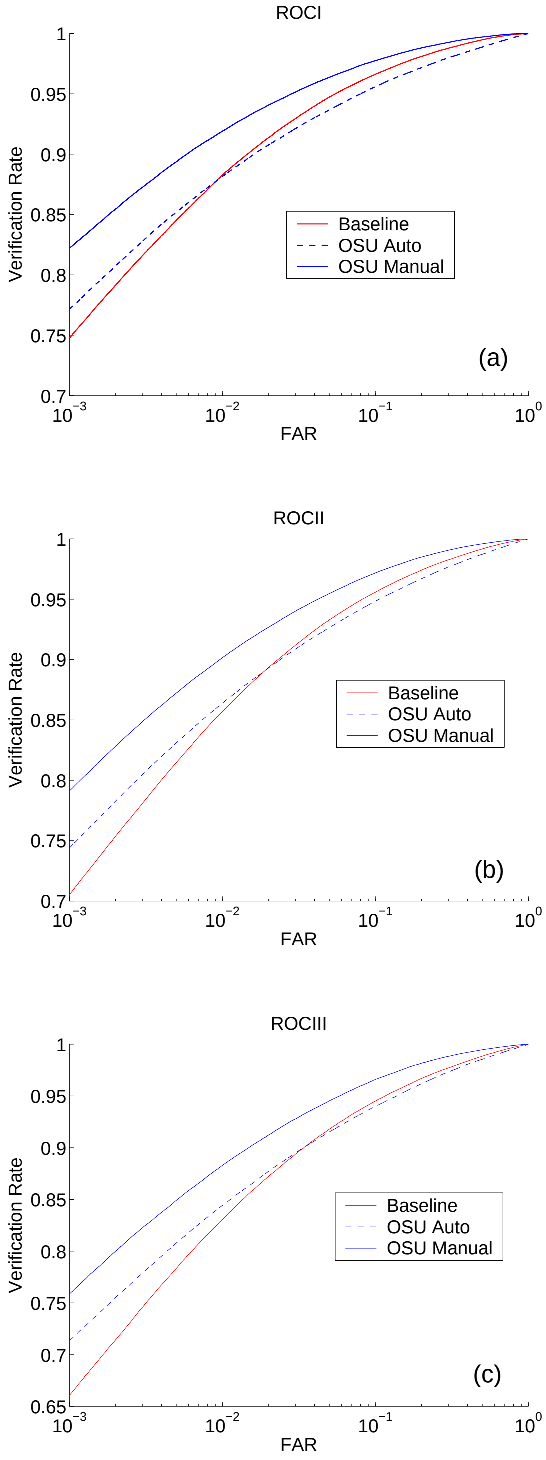


Figure 5: ROC curves corresponding to each of the three sub-tests in Experiment 1. (a) Results for images taken within semesters. (b) Images taken within and between semesters. (c) Images taken between semesters only. The result of the baseline algorithm (PCA), OSU Auto (Ohio State University – using our facial feature detector), and OSU Manual (our algorithm using manually localized facial features) are shown.

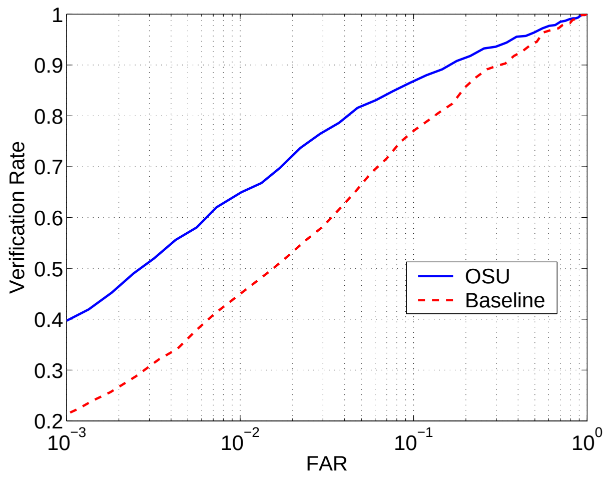


Figure 6: ROC plot for Experiment 4 of Version 1 of the FRGC database.

4.2 Experiment 4

This test incorporates a new dimension to the problems addressed in experiment 1. The images of the query set correspond to those captured in an environment with uncontrolled illumination [11]. Our goal will now be to show whether the local approach defined in this paper is more robust to these changes than the global PCA algorithm used as baseline.

Fig. 6 shows the ROCs of our method and the baseline PCA. Here, we show results using version 1 of the database and using manual detection of the facial features. We see that in this case our algorithm is also superior to those obtained by the baseline algorithm in BEE.

5 Conclusions

This paper presents a study of a probabilistic, part-based framework within the 2005 FRGC test. Our goal was to show that our method successfully addresses the problem posed by errors of localization (when automatic detection is used), and that our algorithm is less sensitive to local changes (as for example, those given by expressions) than a global PCA approach. We have also defined a new method for the automatic detection of facial features, and we have formally justified the use of the Mahalanobis cosine distance.

Acknowledgments

This research was partially supported by NIH under grant R01 DC 005241. We thank Stan Ahalt and the Ohio Super Computer Center for allowing us to use their facility.

References

[1] P.N. Belhumeur and D.J. Kriegman, "What is the Set of Images of an Object Under All Possible Lighting Conditions," Int. J. Comp. Vision, 1998.

[2] R. Beveridge, D. Bolme, M. Teixeira, and B. Draper, "The CSU face identification evaluation system users guide: Version 5," Technical Report, Colorado State University, May 2003.

[3] C. Darwin, "The expression of the emotions in man and animals," London:John Murray, 1872. Re-printed by *The University of Chicago Press* 1965.

[4] P. Ekman and W. Friesen, "Facial Action Coding System: A technique for the measurements of facial movements," Consulting Psychologists Press, 1978.

[5] K. Fukunaga, "Introduction to Statistical Pattern Recognition (second edition)," Academic Press, 1990.

[6] A.S. Georghiades, P.N. Belhumeur and D.J. Kriegman, "From Few to Many: Generative Models of Object Recognition," IEEE Transactions Pattern Analysis and Machine Intelligence 23(6):643-660, 2001.

[7] K. V Mardia, and P. E. Jupp , "Directional Statistics," John Wiley, 2000.

[8] A.M. Martínez, "Recognizing Imprecisely Localized, Partially Occluded and Expression Variant Faces from a Single Sample per Class," IEEE Transactions on Pattern Analysis and Machine Intelligence 24(6):748-763, 2002.

[9] A.M. Martínez, "Matching Expression Variant Faces," Vision Research 43(9):1047-1060, 2003.

[10] A.M. Martínez and Y. Zhang, "Subset Modelling of Face Localization Error, Occlusion, and Expression," In Face Processing: Advance Modeling and Methods (Eds. R. Chellappa & W. Zhao), Academic Press, 2005.

[11] P.J. Phillips, P.J. Flynn, T. Scruggs, K.W. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, "Overview of the Face Recognition Grand Challenge," In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2005.

[12] C.R. Rao, "Linear Statistical Inference and Its Applications," John Wiley, 1973.

[13] N.Ramanathan, A. Roy Chwodhury, R. Chellappa, "Facial Similarity Across Age, Disguise, Illumination and Pose," International Conference on Image Processing, 2004.

[14] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," J. Opt. Soc. Am. A, 4:519-524, 1987.

[15] K.-K. Sung, T. Poggio, "Example-based learning for view-based human face detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(1):39–51, 1998

[16] M. Turk and A. Pentland, “Eigenfaces for recognition,” Journal Cognitive Neuroscience 3(1):71-86, 1991.

[17] M.-H. Yang, D.J. Kriegman and N. Ahuja, “Face Detection Using Multimodal Density Models,” Computer Vision and Image Understanding, 84:264-284, 2001.

[18] W. Zhao, R. Chellappa, J. Phillips and A. Rosenfeld, “Face Recognition in Still and Video Images: A Literature Surrey,” ACM Computing Surveys 35(4):399-458, 2003.

[19] M. Zhu and A.M. Martinez, “Optimal Subclass Discovery for Discriminant Analysis,” In Proceedings of the IEEE Workshop on Learning in Computer Vision and Pattern Recognition, 2004.