Recognition of Partially Occluded and/or Imprecisely Localized Faces Using a Probabilistic Approach

Aleix M. Martínez

Robot Vision Lab, School of Electrical and Computer Eng., Purdue University, IN 47907
Sony Computer Science Lab; 6, rue Amyot, 75005 Paris
aleix@ecn.purdue.edu

Abstract

New face recognition approaches are needed, because although much progress has been recently achieved in the field (e.g. within the eigenspace domain), still many problems are to be robustly solved. Two of these problems are occlusions and the imprecise localization of faces (which ultimately imply a failure in identification). While little has been done to account for the first problem, almost nothing has been proposed to account for the second. This paper presents a probabilistic approach that attempts to solve both problems while using an eigenspace representation. To resolve the localization problem, we need to find the subspace (within the feature space, e.g. eigenspace) that represents this error for each of the training image. To resolve the occlusion problem, each face is divided into $n$ local regions which are analyzed in isolation. In contrast with other previous approaches, where a simple voting space is used, we present a probabilistic method that analyzes how “good” a local match is. Our method has proven to be superior to a local voting PCA on a set of 2600 face images.

1 Introduction

The increasing interest in the Face Recognition (FR) field is fueled by its many potential applications as well as for its scientific interest. Recent work on subspace analysis techniques to FR has proven to be very useful under certain “constrained” domains (e.g. where the illumination conditions remain closely the same for learning and testing shots, where facial expressions between learning and testing do not differ much, etc.) [2]. Among them, it is worthwhile to emphasize the PCA approach, e.g. [5, 13, 8, 9].

Unfortunately, still many problems remain to be solved before the PCA approach can be applied to “unconstrained” domains. Two of these problems are: (i) the localization error and (ii) occlusions. This paper focuses on solving these two issues. The problem we tackle here can be formulated as follows: how can we recognize a face where the localization step does not guarantee a precise localization and/or where parts of the face appear occluded on account of glasses or clothing?

Problem (i) (namely, the localization error problem) is encountered when a small localization error makes the eigen-representation of a test image close to an incorrect class; this is depicted in Figure 1(a). In this figure, we display two classes, one drawn using crosses and another by means of pentagons. For each class, there are two learned feature vectors, each corresponding to the same image but accounting for different localization errors. 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. I.e. while there is a localization that leads to a correct classification, the other does not. This point becomes critical when the learning and testing images differ on facial expression, illumination conditions, etc. as well as for duplicates. A “duplicate” is an image of a face that is taken at a different time, weeks, months or even years later. Note that the localization problem means not a failure in localizing faces: we are assuming that the face localization step succeed, however small errors of precision can make the identification process fail.

We resolve this problem, by learning the subspace that represents small localization errors within the eigenspace. To do that, we first need to calculate the actual error (the physics to be modeled) of our localization system (about the $z$ and $y$ image axes). Second, accounting for this localization error, we project all possible morphed faces onto the eigenspace. And third, for each person the subspace where all his/her possible localization lie are modeled by means of a Gaussian distribution. This process is displayed in Figure 1(b).

Problem (ii) (namely, the occlusion problem) is visu-
Figure 1. (a) The localization problem: different localization lead to different representations onto the feature space, which can cause identification failures. (b) We learn the subspace where different localization lie.

Figure 2. (a) A possible learning image. (b) A test image. (c) The local approach.

for two reasons: it allows replication and it helps us to better describe both the localization error problem and the physics behind our method. The reader should keep in mind that many (localization algorithm) alternatives exist.

2.1 Face localization

Color has now been largely used to localize faces in an image, e.g. [14]. Human skin color forms a dense manifold in color space which makes it an easy feature to detect in images. We use the three dimensional chromatic space (i.e. the normalized $R, G, B$ space $(R/T, G/T, B/T)$, where $T = R + G + B$) to pre-localize faces in our images, Figure 4(b) shows a result. The erosion and dilatation operators from the mathematical morphology are used to get rid of small isolated segments, Figure 4(c). A $4 \times 6$ operator was used. Then a PCA technique is used to localize the eyes and the mouth. Using the differences between the $x$ and $y$ coordinates of the two eyes, the original image is rotated until obtaining a frontal face i.e. both eyes have the same $y$ value (mathematically, $\tan(\angle y_1, y_2/\angle x_1, x_2)$), where $(x_1, y_1)$ and $(x_2, y_2)$ are the right and left eye coordinates). Finally a contour technique is used to search for the boundaries of the face (i.e. top, down, left and right). Figure 4(d) depicts a face localization result. The face is then morphed to a final “standard” face of 120 by 170 pixels, Figure 4(e). After morphing, the eye centers, the medial line of the nose, etc., are expected to be at the same pixel coordinates in all images. Obviously, this will only be true where the localization step has no error, which is almost never the case (this is why, we need to deal with the localization problem). The morphing procedure is necessary, so as to guarantee that only the appearance of the face is taken into account by the recognition system [7].

Notice that not all features can be detected in images (h-m) of Figure 3. To overcome this, the eye posi-
Figure 3. Images of one subject in the AR face database. The images (a) through (m) were taken during one session and the images (n) through (z) at a different session (14 days later).

2.2 The localization error

It would be unrealistic to hope for a system that does localize the above mentioned features (e.g. eyes, mouth, top, etc.) with a high degree of precision. Due to different facial expressions, lighting conditions, etc., the localization stage is expected to be error prone. The trouble with this is that the eigen-representation of the correct localized face differs from the eigen-representation of the actual computed localization, which can ultimately imply an incorrect recognition. An example was depicted in Figure 1(a).

Fortunately, we can model our problem by means of a probability density function. Assume (for the time being) that the localization error (for both, the x and y axes) is known. Then, by simply creating all possible localizations (accounting for this error) in the learning stage, the subspace where this data lies can be easily modeled, Figure 1(b). However, we still need to know the error that our localization algorithm is performing. To do that, we manually localized each of the above enumerated facial features in 500 images. While doing that, we took care to be as precise as possible. Then, we computed the error of the above described localization algorithm in contrast to our manually localized features. The localization was found to have a variance error of ±3 pixels about the x image axis (let us denote this error as $\sigma_r$) and of ±4 pixels about the y image axis (denoted as $\sigma_y$).

Once the localization error is known, the method works as follows. For every sample image, we compute the above described localization approach. Each sample image is then morphed to r possible solutions, which account for all possible errors from 0 to ±3 about the x axis and from 0 to ±4 about the y axis, where r is of the order of $2^{2f}$, being $f$ the number of features localized in each face (precisely, $r = (\sigma_r * \sum_{i=1}^{f} (\frac{1}{i})) * (\sigma_y * \sum_{i=1}^{f} (\frac{1}{i})) = \sigma_r * (2^f - 1) * \sigma_y * (2^f - 1)$). When the number of features detected in an image is large, this complexity is to be considered. In those cases, knowing that accounting for the error in only one feature at a time would not imply a significant change of appearance (in practice, almost none), a subset of all these $r$ is expected to suffice.

3 Local Probabilistic Approach

One way to deal with partially occluded objects (such as faces) is by using local approaches [3, 8, 10]. In such techniques, a face is divided into several parts that are analyzed in isolation, and then a voting space is used to search for the best match. However, a voting technique can easily miss-classify a test image, because
it does not take into account how good a local match is (in relation to other local matches). We present a probabilistic approach that attempts to solve this problem.

3.1 Learning stage

The learning procedure can be divided into two steps: the local eigenspace representation, and the generation of the Gaussian models.

To generate the local eigenspaces, we first apply the localization stage to each of the training images, and then generate the learning set \( \mathbf{x}_k = \{x_{1,k}, \ldots, x_{m,k} \} \), where \( x_{i,j,k} \) is the \( k \)-th local area of the \( j \)-th (morphed) sample image of class \( i \) (in its vector form), \( n \) is the total number of classes (i.e., people), and \( m \) the number of samples per class. In this contribution, we assume that the number of samples is equal for all classes (in any other case, derivations might slightly differ from the ones described below). We also assume \( k = 1, 2, 3, 4, 5, 6 \); i.e., six local areas. To obtain each \( x_{i,j,k} \) sample an ellipse-shape segment \( x^2/d_x^2 + y^2/d_y^2 = 1 \) is used (Figure 2(c)).

After finding the mean vector \( \mu_k \) of each \( x_k \), the covariance matrices \( \mathbf{Q}_k = (x_k - \mu_k)(x_k - \mu_k)^T \), \( \forall k \) are computed. Eigenspaces are obtained by taking the \( p \) eigenvectors associated with the largest eigenvalues of \( \mathbf{Q}_k \). (Note, we are not taking the localization error into account yet.)

The above eigenspace representation is not adequate for recognition where the illumination intensity, illumination sources and/or any other environmental feature changes [1]. Pentland et al. [11] showed that the first three eigenvectors of the eigenspace (normally) correspond to the illumination change. Unfortunately, there are two main problems that make this method impractical. First, we do not know if the database we are given contains different illumination sources/conditions or not. Second, although these eigenvectors basically contain illumination information, they might also contain discriminant information that is essential to discriminate between some of the classes. A better approach is to use a separate eigenspace for each group of images that were taken under the same conditions, i.e., same illumination conditions, same facial expression, etc. To clarify this, pay attention to Figure 3: images (a) to (m) of a single person correspond to 13 different conditions, whereas image (a) for all subjects correspond to a single condition. Note that this approach needs all learning images to be labeled (i.e., to which class they belong to) which might be seen as a drawback.

More formally, let \( \mathbf{x}_{j,k} = \{x_{1,j,k}, \ldots, x_{n,j,k} \} \) be the set of images corresponding to \( k \)-th local part and \( j \)-th learning condition. We can construct all \( j \times k \) eigenspaces by finding the \( p \) eigenvectors associated to the highest eigenvalues of each covariance matrix \( \mathbf{Q}_{i,j,k} = (x_{i,j,k} - \mu_{i,j,k})(x_{i,j,k} - \mu_{i,j,k})^T \); where \( \mu_{i,j,k} \) is the mean feature vector of \( x_{i,j,k} \). In the following, we shall refer to these eigenspaces as \( \mathbf{E}_{i,j,k} \), and to their projecting matrices as \( \mathbf{E}_{i,j,k} \) (which are built by means of the \( p \) eigenvectors associated to the largest eigenvalues of their associated \( \mathbf{Q}_{i,j,k} \)).

Once all those eigenspaces have been computed, we are ready to search for the subspaces that account for the localization error. To achieve that, we need to project all learning instances (accounting for the localization error) onto the above computed eigenrepresentation. Mathematically speaking, we define \( \tilde{\mathbf{x}}_{i,j,k} = \{\tilde{x}_{1,i,j,k}, \ldots, \tilde{x}_{n,i,j,k} \} \), where \( \tilde{x}_{i,j,k} = \{X_{i,j,k}^2, \ldots, X_{i,j,k}^n \} \) and represents all possible images accounting for any possible error of localization (recall that our localization error has a variance of \( \pm 3 \) by \( \pm 4 \) pixels). For obvious reasons, each \( \tilde{\mathbf{x}}_{i,j,k} \) is only projected onto its corresponding eigenspace by means of \( \mathbf{E}_{i,j,k} \). Each \( \tilde{\mathbf{x}}_{i,j,k} \) set is expected to be within a small subspace of its corresponding \( \mathbf{E}_{i,j,k} \), which can be modeled by means of a Gaussian distribution \( \mathcal{G}_{i,j,k} \) with an associated mean \( \mu_{i,j,k} \) and covariance matrix \( \Sigma_{i,j,k} \); where \( \mathcal{G}_{i,j,k} \) is the Gaussian model associated with the training sample \( \tilde{x}_{i,j,k} \), and \( \Sigma_{i,j,k} = (\tilde{x}_{i,j,k} - \mu_{i,j,k})(\tilde{x}_{i,j,k} - \mu_{i,j,k})^T \). Notice that the set \( \tilde{x}_{i,j,k} \) (which can be very large) needs not be stored in memory, only the Gaussian model (mean and covariance matrix) is needed for consecutive computations.

3.2 Identification stage

When a test image \( \mathbf{z} \) is to be recognized, we work as follows. We first localize the face by means of our earlier described method (Section 2), and morph the face to its final 120 by 170 pixel array (which we shall denote as \( \tilde{\mathbf{z}} \)). Since the localization error was already
considered in the learning stage, we need not worry about this problem now.

We project each of the six local areas onto the above computed eigenspaces. More formally, \( \tilde{t}_{z,j,k} = E_{j,k} \cdot \tilde{z}_{j,k}; \ v_{j,k} \) where \( \tilde{z}_{j,k} \) represents the \( k \)th local part of \( \tilde{z} \), and \( \tilde{t}_{z,j,k} \) its projection onto \( E_{i,j,k} \). Notice that since we do not know to which of the environmental conditions this test image belongs, we need to project \( \tilde{t}_{z,j,k} \) onto all possibilities \( (E_{i,j,k}; v_{j,k}) \) and then either search for the best match or add all possibilities (distances). As stated in the introduction, it is better to add probabilities rather than making hard classification decisions, because it allows us to select the correct class even if this is not the best (first) one. Since the mean feature vector and the covariance matrix of each local subspace are already known, the probability of a given match can be directly associated with a suitably defined distance \( (\tilde{t}_{z,j,k} - \mu_{z,j,k}) \Sigma_{i,j,k}^{-1} (\tilde{t}_{z,j,k} - \mu_{z,j,k}) \) i.e. the Mahalanobis distance. Mathematically, \( \text{LocRes}_{i,k} = \sum_{j=1}^{m} \text{MH}_i(\tilde{t}_{z,j,k}, \hat{G}_{i,j,k}) \) where \( \text{MH}_i(\cdot) \) is the Mahalanobis distance, and \( \text{LocRes}_{i,k} \) the recognition result of the \( k \)th local area of class \( i \). Generally, the distance from the test sample vector to each \( \hat{E}_{i,j,k} \) is also taken into account; i.e. added to \( \text{LocRes}_{i,k} \).

Finally, we add all local distances (probabilities), \( \text{Res}_i = \sum_{k=1}^{6} \text{LocRes}_{i,k} \), and search for the minima (maxima). \( \text{RecClass} = \text{argmin}_i \text{Res}_i \); where \( \text{RecClass} \in [1, \overline{n}] \). If a video sequence is supplied, we keep adding distances (probabilities) for each of the images and only compute the minima (maxima) at the end of the sequence or when a threshold has been reached.

4 Experimental Results

The results to be presented in this section were obtained using the AR-face database [6]. This database consists of over 3200 color images of the frontal images of faces of 126 subjects. There are 26 different images for each subject. For each person, these images were grabbed in two different sessions separated by two weeks, 13 images being recorded in each session. For illustration, these images for one subject were shown in Figure 3. All images were taken using the same camera under tightly controlled conditions of illumination. Each image in the database consists of a 768 \( \times \) 576 array of pixels, and each pixel is represented by 24 bits of RGB color values. For the experiments reported in this section, 100 different individuals (50 males and 50 females) were randomly selected from this database (that is a total of 2600 images). As stated earlier, images were morphed to a final 120 \( \times \) 170 pixel size array.

After morphing, all images were converted to gray-level images by adding all three color channels.

For each of the experimental results a set of images was assigned to learning (denoted as gallery) and another to testing (denoted as probes). The following table summarizes our tests:

<table>
<thead>
<tr>
<th>Test number</th>
<th>Gallery</th>
<th>Probes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(a) / 100</td>
<td>(b-d,h,k) / 500</td>
</tr>
<tr>
<td>2</td>
<td>(a) / 100</td>
<td>(n-q,u,x) / 600</td>
</tr>
<tr>
<td>3</td>
<td>(a-b) / 200</td>
<td>(c-d,h,k) / 400</td>
</tr>
<tr>
<td>4</td>
<td>(a-b) / 200</td>
<td>(o-q,u,x) / 500</td>
</tr>
<tr>
<td>5</td>
<td>(a,g) / 200</td>
<td>(b-f,h-m) / 1100</td>
</tr>
<tr>
<td>6</td>
<td>(a,g) / 200</td>
<td>(u-z) / 1300</td>
</tr>
</tbody>
</table>

Values within the Gallery and Probes boxes state for: (image names as in Figure 3) / quantity of images employed. For all these tests, 20-dimensional eigenspaces were used. The local ellipsoidal areas were of \( d_{x} = 30 \) \( d_{y} = 24 \). Results of test 1 are displayed in Fig. 5(a). These results are shown for each separate testing image to facilitate discussion. Figure 5(b-c) summarizes results of tests 1 to 6. In these cases, results are shown as: Local PCA: results of our method (shown as hard lines); Voting PCA: results obtained using a voting method to select the best local match, i.e. \( \text{LocRes}_{c,k} = 1 \), where \( c = \text{argmin}_i \text{MH}(\tilde{t}_{z,j,k}, \hat{G}_{i,j,k}) \)—note we are still using our probabilistic method in all other steps—(this is shown as dotted lines); Global PCA: results of the PCA approach as described in [15] (displayed with open symbols). Results are shown as a function of Rank (which means that the correct solution is within the \( R \) nearest outcomes) and Cumulative match score (which refers to the percentage of successfully recognized images) [12].

From Figure 5(a), we can appreciate that our method easily handles occlusions due to scarring—i.e. where the mouth’s part of the image is not used to identify the test image—(85 \( \sim \) 95\%) and partially solve the sunglasses occlusion problem (\( \sim \) 80\%). Similar results were obtained in [3] (see Fig. 17 in pg. 1049), where the use of the mouth area obtained the worst results (in comparison to the eyes and the nose).

We can also see (Figure 5(a)) that our method is quite robust to the recognition of small changes due to facial expressions (around 84\% for smiles and 94\% for anger). This is mainly due to the fact that although some of the local parts tend to change widely, others remain similar. It is also likely that although small local changes make the eigen-representation of a test face to be close to an incorrect class, the correct match is still quite probable (thus, adding valuable information to the global recognition). It is also to be mentioned the rapid increasing match score of the scream face (e.g. Figure 3(d)), moving from \( \sim \) 50\% for Rank = 1 to \( \sim \) 90\% for Rank = 19. Even though incorrect
classes are chosen for such extreme facial expressions, still good information is available.

It is clear, for our data-sets at least (Figure 5(b-c)), that our new probabilistic approach outperforms a voting method that also accounts for the localization and occlusion problems. The comparison with global PCA is unfair (since this approach does not attempt to solve any of the problems here analyzed), and is only shown as a reference. The recognition of duplicate images was also expected to improve, because some local areas are expected to change less than others. The results displayed in Figure 5(c) confirmed our expectations.

5 Conclusions

In this communication, we have presented a new approach to FR that is robust to imprecise localization of the mug face and allows faces to appear under certain occlusions. Our method has also shown to be capable of learning local features (such as illuminations) that can (in some cases) be very useful for recognition. The probabilistic approach described has shown to be superior to a local PCA method where a hard classification is used to select for the best match in each local area. The use of other PCA techniques such as [9], is of interest for future research.

Acknowledgements


References


