Face recognition using OPRA -faces

E. Kokiopoulou and Y. Saad
Comp. Sci. & Eng. Dept.
University of Minnesota
{kokiopou,saad}@cs.umn.edu

***Abstract*— This paper presents a method named "Orthogonal Projection Reduction by Affinity", or OPRA -faces, for face recognition. As its name indicates, the method consists of an (explicit) orthogonal mapping from the data space to the reduced space. In addition, the method attempts to preserve the local geometry, i.e., the affinity of the points in a geometric representation of the data. The method starts by computing an *affinity mapping* W, of the data, which optimally expresses each point as a convex combination of a few nearest neighbors. This mapping can be viewed as an optimal representation of the intrinsic neighborhood geometries and is computed in a manner that is identical with the method of Locally Linear Embedding (LLE). Next, and in contrast with LLE, the proposed scheme computes an explicit linear mapping between the high dimensional samples and their corresponding images in the reduced space, which is designed to preserve this affinity representation W. OPRA -faces shares some properties with Laplacianfaces, a recently proposed technique for face recognition, which computes the linear approximation of the Laplace-Beltrami operator on the image manifold. Laplacianfaces aims at preserving locality but does not explicitly consider the intrinsic geometries of the neighborhoods as does OPRA. As a result of the preservation of the affinity mapping W, OPRA will tend to produce a linear subspace which captures the essential geometric characteristics of the dataset. This feature, which appears to be crucial in representing images, makes the method very effective as a tool for face recognition. OPRA is tested on standard face databases and its effectiveness is compared with that of Laplacianfaces, Eigenfaces and Fisherfaces. The experimental results indicate that the proposed technique produces results that are sharply superior to the other methods, at a comparable or lower cost.**

I. Introduction

Face recognition [16] is one of the most challenging problems in computer vision and has numerous applications ranging from security to contactless human-machine interaction. One of the most successful class of methods for face recognition is the class of *appearance-based* [16] methods. In these methods, facial images are often represented lexicographically as vectors in a high dimensional space and a lower dimensional linear subspace which captures certain properties of the dataset is constructed. Then, linear dimensionality reduction is invoked in order to project both training and test data in the lower dimensional space. Recognition is then performed among the projected data in the reduced space using a simple classifier such as the nearest neighbor classifier. Well known methods of this class include Eigenfaces [8], [10], [14], Fisherfaces [1] and Laplacianfaces [6], [7], [5].

Work supported by NSF under grant DMS-0510131 and by the Minnesota Supercomputing Institute.

Eigenfaces employ Principal Component Analysis (PCA) [15] for constructing the linear subspace, which is usually called face space. PCA aims at preserving the global structure and seeks orthogonal axes of maximum variance. These are obtained by computing the principal eigenvectors of the sample covariance matrix. PCA is appropriate when the data samples (approximately) lie on a linear subspace. However, it has been observed that the manifold of facial images is intrinsically nonlinear [11] and this can render PCA ineffective in capturing facial images manifolds.

The method of Fisherfaces employs Linear Discriminant Analysis (LDA) [15] for computing the dimensionality reduction matrix. Its basis is to compute a set of directions which are optimal for discriminating information. These directions are obtained by solving a generalized eigenproblem and as a result they are not mutually orthogonal.

Note that both PCA and LDA consider only the Euclidean structure and do not take into account the data topology. Recently, a method named Laplacianfaces was introduced which models explicitly the data topology, by means of a weighted graph. It was shown [6] that Laplacianfaces is able to capture the nonlinear structure of the image manifold and yield an effective method for face recognition. Laplacianfaces builds a linear subspace for face representation, which is designed to preserve the locality of the data samples. Similarly to Fisherfaces, the dimensionality reduction matrix is obtained by solving a generalized eigenproblem which involves the Laplacian matrix of the graph and hence, the resulting axes are not mutually orthogonal.

OPRA -faces, the method proposed in this paper, also models explicitly the data topology by a weighted graph. The graph used by OPRA -faces expresses in some least-squares sense, each point as a convex combination of a few nearest neighbors. This weighted graph can be viewed as an optimal representation of the intrinsic neighborhood geometries and it is computed in a manner that is identical with the method of Locally Linear Embedding (LLE) [11], [13]. In this paper we refer to this graph as the *affinity graph*. A major difference with the standard LLE where the mapping between the input and the reduced spaces is implicit, is that OPRA -faces is an appearance-based method which employs an explicit linear mapping between the two. The embedding of LLE is defined only on the training points and it is cumbersome to extend it to handle new data samples (see, e.g. research efforts by Bengio et al [3]). In contrast, treating new data samples is straightforward in our algorithm, as this amounts to a simple linear transformation.

The proposed method shares some properties with Laplacianfaces, since they both rely on a k-nearest neighbor graph in order to capture the data topology. However, our algorithm inherits the characteristics of LLE in preserving the geometric structure of local neighborhoods, while Laplacianfaces aims at preserving only locality without specifically aiming at preserving the geometry. Experiments suggest that it is important to try to preserve the affinity graph when one uses nearest neighbor classifiers for recognition in the reduced space. An additional advantage of OPRA -faces is that it employs mutually orthogonal axes in contrast with Laplacianfaces, where the projection axes are not orthogonal. OPRA -faces is able to capture the nonlinear structure of the manifold and experimental evidence suggests that it is an effective method for face recognition.

II. Related work

This section gives a brief review of the most representative related methods: Eigenfaces [8], [10], [14], Fisherfaces [1], and Laplacianfaces [6], [7], [5]. Consider a collection of facial images represented by the columns of a matrix $X = [x_1, x_2, \dots, x_n] \in R^{m\times n}$, where the i-th image is lexicographically represented by the data sample x_i. All the above methods are characterized by a common framework. First, they compute a dimensionality reduction matrix V. Next, this matrix is used for projecting the data samples onto the reduced space by computing $y_i = V^\top x_i,\ i = 1, \dots, n$. Finally, recognition is performed in the reduced space (among y_i's) using a simple classifier. The methods are differentiated by the way in which the matrix V is determined.

The method of Eigenfaces employs PCA to determine V. In PCA, the matrix V is computed such that the variance of the projected vectors is maximized i.e, $\max_{V\in R^{m\times d}} \left\| y_i - \frac{1}{n}\sum_{j=1}^{n} y_j \right\|_2^2$, under the orthogonality constraints $V^\top V = I$. It turns out that the column vectors of the solution V to this problem are the principal eigenvectors of the sample covariance matrix [15].

Fisherfaces determines V by using Linear Discriminant Analysis (LDA). LDA works by extracting a set of "optimal" discriminating axes. Assume that we have c classes and that class i has n_i data points. Define the *between-class scatter matrix* $S_B = \sum_{i=1}^{c} n_i(\mu^{(i)} - \mu)(\mu^{(i)} - \mu)^\top$ and the *within-class scatter matrix* $S_W = \sum_{i=1}^{c}\left(\sum_{j=1}^{n_i}(x_j^{(i)} - \mu^{(i)})(x_j^{(i)} - \mu^{(i)})^\top\right)$ where $\mu^{(i)}$ is the centroid of the i-th class. In LDA the columns of V are the eigenvectors associated with largest eigenvalues of the generalized eigenvalue problem $S_B w = \lambda S_W w$.

Laplacianfaces [6] constructs the weighted k nearest neighbor (k-NN) graph and builds a similarity matrix S, whose entry S_{ij} represents the edge weight between nodes x_i and x_j. The authors in [6] propose the use of Gaussian weights, where $S_{ij} = e^{-\frac{\|x_i - x_j\|^2}{\sigma}}$, when x_i is among the k nearest neighbors of x_j (or vice versa) and 0 otherwise. The selection of the parameter σ is crucial for the performance of the algorithm. Laplacianfaces employs the following objective function

$$\sum_{ij}(y_i - y_j)^2 S_{ij}, \tag{1}$$

which is identical with that of the method of Laplacian Eigenmaps [2], a nonlinear technique for dimensionality reduction. The main difference with Laplacian Eigenmaps is that Laplacianfaces is linear and employs an explicit linear mapping $X \rightarrow Y$. The objective function (1) captures the locality of the data samples and results in the generalized eigenproblem

$$XLX^\top v = \lambda XDX^\top v, \tag{2}$$

where $D = \sum_i S_{ij}$ and $L = D - S$ is the Laplacian matrix. The eigenvectors of the above problem corresponding to the smallest eigenvalues yield the dimensionality reduction matrix V used by Laplacianfaces.

III. Face Recognition Using OPRA -faces.

The process of OPRA -faces consists of two parts. The first part is identical with that of LLE [11], [13] and consists of computing some optimal weights in each neighborhood. The basic assumption is that each data sample along with its k nearest neighbors (approximately) lies on a locally linear manifold. Hence, each data sample x_i is reconstructed by a linear combination of its k nearest neighbors. The reconstruction errors are minimized via the objective function

$$\mathcal{E}(W) = \sum_i \|x_i - \sum_j W_{ij} x_j\|_2^2. \tag{3}$$

The weight W_{ij} represent the linear coefficient for reconstructing the sample x_i from its neighbors $\{x_j\}$. The following constraints are imposed on the weights:

1) $W_{ij} = 0$, if x_j is not one of the k nearest neighbors of x_i;
2) $\sum_j W_{ij} = 1$, that is x_i is approximated by a convex combination of its neighbors.

Note that the optimization problem (3) can be recast in matrix form as $\min_W \|X(I - W^\top)\|_F$, where W is an $n \times n$ sparse matrix which has a specific sparsity pattern (condition (1)) and satisfies the constraint that its row-sums be equal to one (condition (2)). The weights for a specific data point x_i are computed as follows. Define $C_{pl} = (x_i - x_p)^\top (x_i - x_l) \in R^{k\times k}$, the local Gram matrix containing the pairwise inner products among the neighbors of x_i, given that the neighbors are centered with respect to x_i. It can be shown that the weights of the above constrained least squares problem are given in closed form [11] using the inverse of C,

$$w_i = \frac{\sum_p C_{ip}^{-1}}{\sum_{pl} C_{pl}^{-1}}\ , \tag{4}$$

where w_i represents the i-th column of W. The weights W_{ij} satisfy certain optimality properties. They are invariant to rotations, scalings, and translations. As a consequence of these properties the affinity graph preserves the intrinsic geometric characteristics of each neighborhood.

Algorithm **OPRA-FACES**
Input: Dataset $X \in R^{m\times n}$ and d: dimension of reduced space, k: number of NN, c: number of classes and ℓ: class labels.
Output: Dimensionality reduction matrix $V \in R^{m\times d}$ and projected vectors $Y = [y_1, y_2, \ldots, y_n] \in R^{d\times n}$.
1. Employ PCA projection on X to reduce dimension to $n - c$. Call V_{PCA} the dimensionality reduction matrix of PCA.
2. Compute the k nearest neighbors of each data point $x_i,\ i = 1, \ldots, n$.
3. Compute the weights using equation (4) that give the best linear reconstruction of each data point x_i by its neighbors.
4. Compute the projected vectors $y_i = V^\top x_i,\ i = 1, \ldots, n,$ where $V = V_{\mathrm{PCA}}V_{\mathrm{OPRA}}$ and V_{OPRA} is determined by computing the $d + 1$ eigenvectors of $\tilde{M} = X(I - W^\top)(I - W)X^\top$ corresponding to its smallest eigenvalues. The smallest eigenvector is ignored.

TABLE I
THE OPRA -FACES ALGORITHM.

Consider now the second part of projecting the data samples X to the reduced space $Y = [y_1, y_2, \ldots, y_n] \in R^{d\times n}$. OPRA - faces imposes an explicit linear mapping from $X \to Y$ such that $y_i = V^\top x_i,\ i = 1, \ldots, n$ for an appropriately determined matrix $V \in R^{m\times d}$. In order to determine the matrix V, OPRA -faces imposes the constraint that each data sample y_i in the reduced space is reconstructed from its k neighbors by exactly the same weights as in the input space. This leads to the solution of the following optimization problem, where we set $M = (I - W^\top)(I - W)$ and $\tilde{M} = XMX^\top$

$$\begin{aligned}\min_Y \Phi(Y) &= \min_Y \sum_i \|y_i - \sum_j W_{ij} y_j\|_2^2 \\ &= \min_{V\in R^{m\times d}} \sum_i \|V^\top x_i - \sum_j W_{ij} V^\top x_j\|_2^2 \\ &= \min_{V\in R^{m\times d}} \|V^\top X(I - W^\top)\|_F^2 \\ &= \min_{V\in R^{m\times d}} \mathrm{tr}(V^\top XMX^\top V) \\ &= \min_{V\in R^{m\times d}} \mathrm{tr}(V^\top \tilde{M} V)\ . \end{aligned} \tag{5}$$

If we impose the additional constraint that the columns of V are orthonormal, i.e. $V^\top V = I$, then the solution V to the above optimization problem is the basis of the eigenvectors associated with the d smallest eigenvalues of $\tilde{M}$. We observed in practice that ignoring the smallest eigenvector of $\tilde{M}$ is helpful. This is an issue to be investigated in future work. Note that the embedding vectors of LLE are obtained by computing the eigenvectors of the matrix M associated with its smallest eigenvalues.

Consider now a new facial test point x_t which must be recognized. The test vector is projected onto the subspace $y_t = V^\top x_t$ using the dimensionality reduction matrix V. Next, it is compared to the training samples $y_i,\ i = 1, \ldots, n$ and recognition is performed using a nearest neighbor (NN) classifier based on the Euclidean distance.

IV. SUPERVISED OPRA -FACES

OPRA -faces can be implemented in either a supervised or an unsupervised setting. In the supervised case where the class labels are available, OPRA -faces can be modified appropriately and yield a projection which carries not only geometric information but discriminating information as well. The method starts by building the affinity graph $G = (N, E)$, where the nodes N correspond to data samples and an edge $e_{ij} = (x_i, x_j)$ exists if and only if x_i and x_j belong to the same class. In other words, we make adjacent those nodes (data samples) which belong to the same class. Notice that in this case one does not need to set the parameter k, the number of nearest neighbors, so the method becomes fully automatic.

Denote by c the number of classes and n_i the number of data samples which belong to the i-th class. The data graph G consists of c cliques, since the adjacency relationship between two nodes reflects their class relationship. This implies that with an appropriate reordering of the columns and rows, the weight matrix W will have a block diagonal form where the size of the i-th block is equal to the size n_i of the i-th class. In this case W will be of the following form,

$$W = \mathrm{diag}(W_1, W_2, \ldots, W_c).$$

The weights W_i within each class are computed in the usual way, see eq. (4). The rank of W defined above, is restricted as is explained by the following proposition.

Proposition 1: The rank of W is at most $n - c$.

Proof: Recall that the row sum of the weight matrix W_i is equal to 1, because of the constraint (2). This implies that $W_i e_i = 0, e_i = [1, \ldots, 1]^\top \in R^{n_i}$. Thus, the following c vectors

$$\begin{bmatrix} e_1 & 0 & \cdots & 0 \\ 0 & e_2 & \cdots & 0 \\ 0 & 0 & \cdots & e_c \end{bmatrix},$$

are linearly independent and belong to the null space of W. Therefore, the rank of W is at most $n - c$. ■

Consider now the case $m > n$ where the number of samples (n) is less than their dimension (m). This case is known as the *undersampled size* problem and occurs very often in face databases. A direct consequence of the above proposition is that in this case, the matrix $\tilde{M} \in R^{m\times m}$ will have rank at most $n - c$. In order to ensure that the resulting matrix $\tilde{M}$ will be nonsingular, we may employ an initial PCA projection that reduces the dimensionality of the data vectors to $n-c$. Call V_{PCA} the dimensionality reduction matrix of PCA. Then the OPRA -faces algorithm is performed and the total dimensionality reduction matrix is given by $V = V_{\mathrm{PCA}}V_{\mathrm{OPRA}}$, where V_{OPRA} is the dimensionality reduction matrix of OPRA -faces. The main steps of the OPRA -faces algorithm are summarized in Table I.

V. DISCUSSION

PCA and LDA are traditional linear techniques which consider only the Euclidean structure. They do not take into account the nonlinear structure of the image manifolds. On the other hand OPRA -faces and Laplacianfaces explicitly model the data structure and topology by means of a weighted k-NN graph. Moreover, PCA and LDA are global methods which do not aim at preserving locality. On the other hand, OPRA - faces and Laplacianfaces aim at preserving local geometry

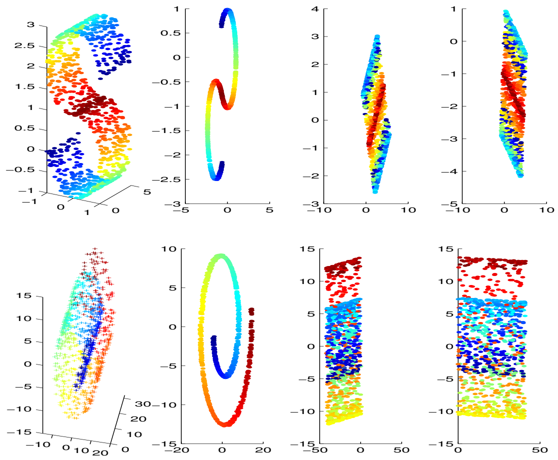

Fig. 1. Results of applying all methods on the `s-curve` and the `swissroll`. From left to right: OPRA-faces, Laplacianfaces and PCA.

and locality respectively. This last feature is very important especially when one performs recognition in the reduced space using NN classifier (as is usually done in appearance-based methods).

OPRA-faces shares some properties with Laplacianfaces. Note that the former inherits the optimal weights from LLE which represent the intrinsic local geometries. In contrast, Laplacianfaces aims at preserving only locality and does not consider the geometric structure explicitly. Thus, the geometric structure of the neighborhoods in the reduced space may be perturbed. Note also that the Gaussian weights used in Laplacianfaces are somewhat artificial and may not reflect the underlying geometry. In addition, the selection of the parameter σ, the width of the Gaussian envelope, is crucial for the performance of the algorithm. This issue is often overlooked, but it is an important weakness associated with the use of Gaussian weights. The supervised version of OPRA-faces is fully automatic. Indeed, the only parameter, the number of nearest neighbors k, is implicitly determined by the corresponding class size n_i. Finally, the dimensionality reduction matrix V of OPRA-faces has orthonormal columns. This is very helpful in preserving angles as much as possible in the reduced space. This is to be contrasted with Laplacianfaces where the matrix V is not orthogonal since its columns are eigenvectors of a generalized eigenproblem.

VI. Experimental results

A. Artificial datasets

We demonstrate the advantageous characteristics of OPRA-faces over the other methods by applying it on two popular artificial datasets: the *s-curve* and *the swissroll* [11], [13]. The results are illustrated in Figure 1. We uniformly sample $n = 1,000$ data points from the `s-curve` and the `swissroll` and the discretized manifold is illustrated in the left panels. The number of neighbors is $k = 12$. Each data point is projected in the two-dimensional space using the corresponding dimensionality reduction matrix V of each algorithm. Observe that OPRA-faces preserve locality (indicated by the color shading) since nearby points in the input space are mapped nearby in the output two dimensional space. In addition, notice that the angles are preserved as much as possible and the projection at the reduced space is faithful and conveys meaningful information about how the manifold is folded in the higher dimensional space.

Fig. 3. Sample face images from the UMIST database. The number of different poses poses for each subject is varying.

B. Face recognition

We used three datasets that are publically available: UMIST [4], ORL [12], and AR [9]. For computational efficiency the images in all databases were downsampled to size 38×31. Thus, each facial image was represented lexicographically as a high dimensional vector of length 1,178. In order to measure the recognition performance, we use a random subset of facial expressions/poses from each subject as training set and the remaining as test set. In order to ensure that our results are not biased from a specific random realization of the training/test set, we perform 20 different random realizations of the training/test sets and we report the average error rate. Figure 2 illustrates the first 10 basis vectors of all methods from the ORL database, in a pictorial fashion. Observe that the basis vectors of Fisherfaces, Laplacianfaces and OPRA-faces are pretty similar and this may be due to the fact that these are supervised methods and their basis vectors encode discriminating information.

Note that in what follows, we test with the supervised version of OPRA-faces (see Section IV for more details) and Laplacianfaces. In the latter algorithm, we employ Gaussian weights. We determine the value of the width σ of the Gaussian envelope as follows. First, we sample 1000 points randomly and then compute the pairwise distances among them. Then σ is set equal to half the median of those pairwise distances. This gives a good and reasonable estimate for the value of σ.

C. UMIST

The UMIST database [4] contains 20 people under different poses. The number of different views per subject varies from 19 to 48. We used a cropped version of the UMIST database that is publically available from S. Roweis' web page[1]. Figure 3 illustrates a sample subject from the UMIST database along with its first 20 views. We form the training set by a random subset of 15 different poses per subject (300 images in total) and use the remaining poses as a test set. We experiment with the dimension of the reduced space $d = [10 : 5 : 70]$ (in MATLAB notation) and for each value of d we plot the average error rate across 20 random realizations of the training/set

[1] `http://www.cs.toronto.edu/~roweis/data.html`

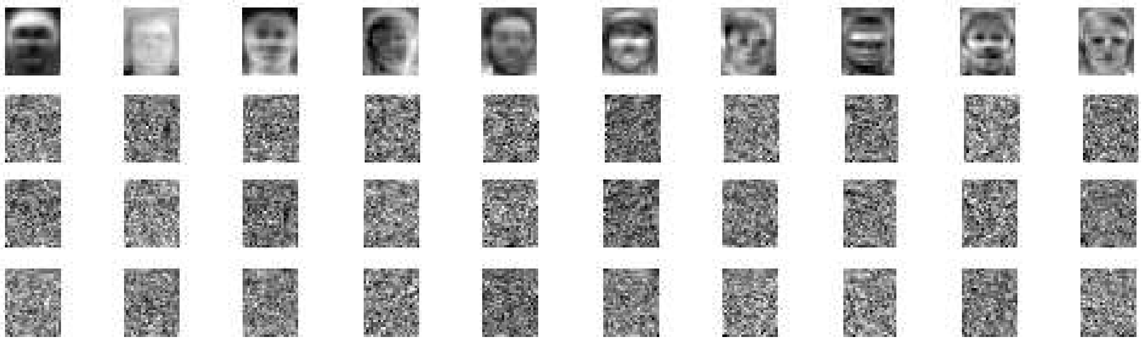

Fig. 2. The first 10 basis vectors from the ORL database illustrated in a pictorial fashion. From top to bottom: Eigenfaces, Fisherfaces, Laplacianfaces and OPRA -faces.

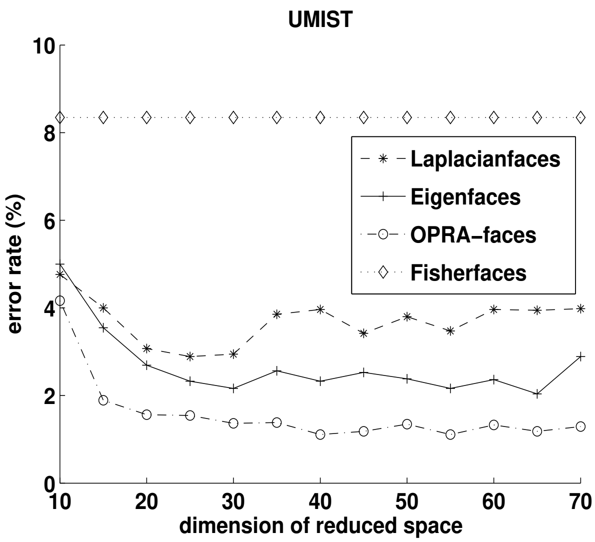

Fig. 4. Error rate with respect to the reduced dimension d on the UMIST database.

	d	error (%)
Eigenfaces	65	2.04
Fisherfaces	14	8.34
Laplacianfaces	25	2.89
OPRA -faces	55	**1.11**

TABLE II

THE BEST ERROR RATE ACHIEVED BY ALL METHODS ON THE UMIST DATABASE.

set. The results are illustrated in Figure 4. Concerning the method of Fisherfaces note that there are only $c-1$ generalized eigenvalues, where c is the number of subjects in the dataset. Thus, d cannot exceed $c-1$ and so we plot only the best achieved error rate by Fisherfaces across the various values of d. Observe that OPRA -faces outperforms the other methods across all values of d. We also report the best error rate achieved by each method and the corresponding dimension d of the reduced space. The results are tabulated in Table II. Both Eigenfaces and OPRA -faces perform very well, with OPRA -faces showing a clear margin of superiority over the other methods.

Fig. 5. Sample face images from the ORL database. There are 10 available facial expressions and poses for each subject.

	d	error (%)
Eigenfaces	30	6.6
Fisherfaces	35	10.37
Laplacianfaces	40	10.47
OPRA -faces	100	**5.83**

TABLE III

THE BEST ERROR RATE ACHIEVED BY ALL METHODS ON THE ORL DATABASE.

D. ORL

The ORL (formerly Olivetti) database [12] contains 40 individuals and 10 different images for each individual including variation in facial expression (smiling/non smiling) and pose. Figure 5 illustrates two sample subjects of the ORL database along with variations in facial expression and pose. We form the training set by a random subset of 5 different facial expressions/poses per subject and use the remaining 5 as a test set. We experiment with the dimension of the reduced space $d = [10 : 10 : 150]$ and for each value of d we illustrate in Figure 6 the average error rate across 20 random realizations of the training set. Observe that for d less than 30, Eigenfaces give the best results, but for all values of d larger than 30 the OPRA -faces method outperforms its counterparts. The best error rates achieved by each method are tabulated in Table III along with the corresponding value of d. Notice that the proposed scheme is superior to Eigenfaces.

E. AR

We use a subset of the AR face database [9] which contains 126 subjects under 8 different facial expressions and variable lighting conditions for each individual. Figure 7 depicts two subjects randomly selected from the AR database under various facial expressions and illumination. We form the training

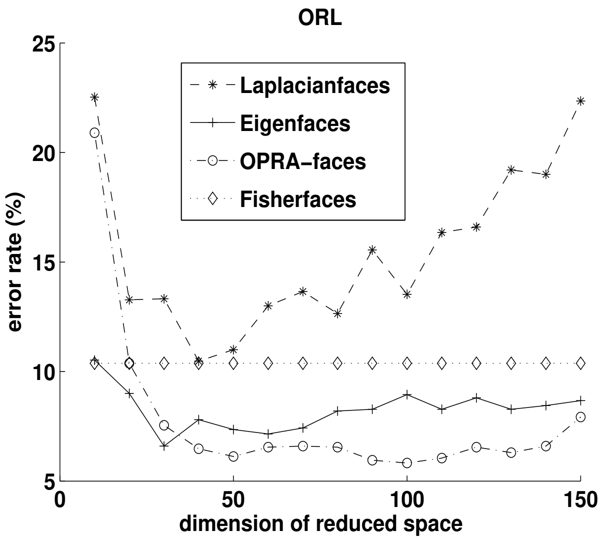

Fig. 6. Error rate with respect to the reduced dimension d on the ORL database.

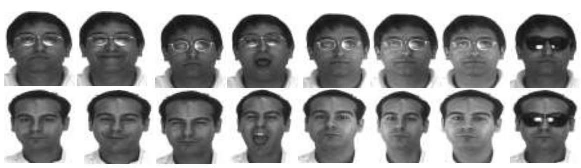

Fig. 7. Sample face images from the AR database. Facial expressions from left to right: 'natural expression', 'smile', 'anger', 'scream', 'left light on', 'right light on', 'all side lights on' and 'wearing sun glasses'.

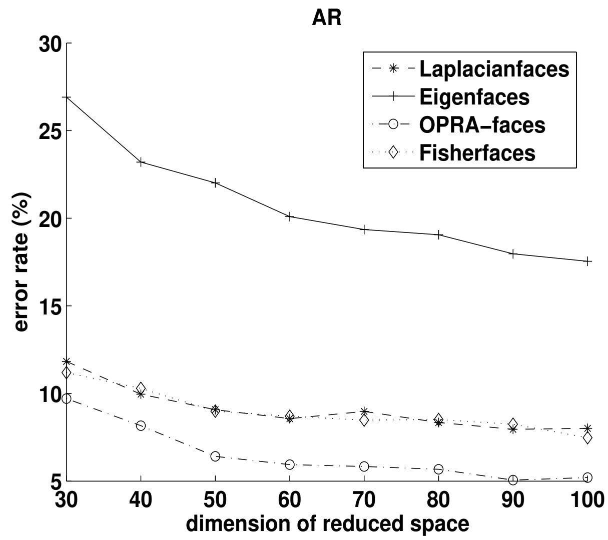

Fig. 8. Error rate with respect to the reduced dimension d on the AR database.

	d	error (%)
Eigenfaces	100	17.53
Fisherfaces	100	7.48
Laplacianfaces	90	7.96
OPRA -faces	90	**5.05**

TABLE IV

THE BEST ERROR RATE ACHIEVED BY ALL METHODS ACROSS ON THE AR DATABASE.

set by a random subset of 4 different facial expressions/poses per subject and use the remaining 4 as a test set. We plot the error rate across 20 random realizations of the training/test set, for $d = [30 : 10 : 100]$. The results are illustrated in Figure 8. Observe that OPRA -faces outperforms its counterparts across all values of d. Also it seems that Laplacianfaces compete with Fisherfaces. Furthermore, Table IV reports the best achieved error rate and the corresponding value of d. Again, OPRA - faces outperforms its competitors.

VII. Conclusion

OPRA -faces, a fully automatic face recognition algorithm aims at preserving the affinity graph, i.e., the local geometries of the data samples in the high dimensional space. The method is able to capture the nonlinear features of the dataset by means of the affinity data graph. OPRA -faces was tested for face recognition using a few well known, and extensively studied, facial databases and was shown to outperforme three popular rival methods on these test cases.

References

[1] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specifi c Linear Projection. *IEEE Trans. Pattern Analysis and Machine Intelligence, Special Issue on Face Recognition*, 19(7):711—20, July 1997.

[2] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput.*, 15(6):1373–1396, 2003.

[3] Y. Bengio, J-F Paiement, P. Vincent, O. Delalleau, N. Le Roux, and M. Ouimet. Out-of-Sample Extensions for LLE, Isomap, MDS, Eigenmaps, and Spectral Clustering. In Sebastian Thrun, Lawrence Saul, and Bernhard Schölkopf, editors, *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.

[4] D. B Graham and N. M Allinson. Characterizing Virtual Eigensignatures for General Purpose Face Recognition. *Face Recognition: From Theory to Applications*, 163:446–456, 1998.

[5] X. He and P. Niyogi. Locality preserving projections. *Advances in Neural Information Processing Systems 16 (NIPS 2003)*, 2003. Vancouver, Canada.

[6] X. He, S. Yan, Y. Hu, P. Niyogi, and H-J Zhang. Face recognition using Laplacianfaces. *IEEE TPAMI*, 27(3):328–340, March 2005.

[7] X. He, S. Yan, Y. Hu, and H-J. Zhang. Learning a locality preserving subspace for visual recognition. *IEEE Int. Conf. on Comp. Vision*, 2003.

[8] M. Kirby and L. Sirovich. Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces. *IEEE Trans. Patt. Anal. Mach. Intell.*, 12, 1990.

[9] A.M. Martinez and R. Benavente. The AR Face Database. Technical report, CVC no. 24, 1998.

[10] A. Pentland, B. Moghaddam, and T. Starner. View-based and Modular Eigenspaces for Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[11] S. Roweis and L. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290:2323–2326, 2000.

[12] F. Samaria and A. Harter. Parameterisation of a Stochastic Model for Human Face Identifi cation. In *2nd IEEE Workshop on Applications of Computer Vision*, Sarasota FL, December 1994.

[13] L. Saul and S. Roweis. Think Globally, Fit Locally: Unsupervised Learning of Nonlinear Manifolds. *Journal of Machine Learning Research*, 4:119–155, 2003.

[14] M. Turk and A. Pentland. Face Recognition using Eigenfaces. In *Int. Conf. on Patt. Recog.*, pages 586–591, 1991.

[15] A. Webb. *Statistical Pattern Recognition*. Wiley, 2nd edition, 2002.

[16] W. Zhao, R. Chellapa, P. Phillips, and A. Rosenfeld. Face Recognition: a Literature Survey. *ACM Computing Surveys*, 35(4):399–458, December 2003.