Subspace Approximation of Face Recognition Algorithms: An Empirical Study

Pranab Mohanty, Sudeep Sarkar, and Rangachar Kasturi
Computer Science and Engineering
University of South Florida, Tampa, Florida 33620, USA
Email: {pkmohant, sarkar, r1k}@cse.usf.edu
P. Jonathon Phillips
National Institute of Standards and Technology
NIST, 100 Bureau Dr.
Gaithersburg, MD 20899
Email: jonathon@nist.gov

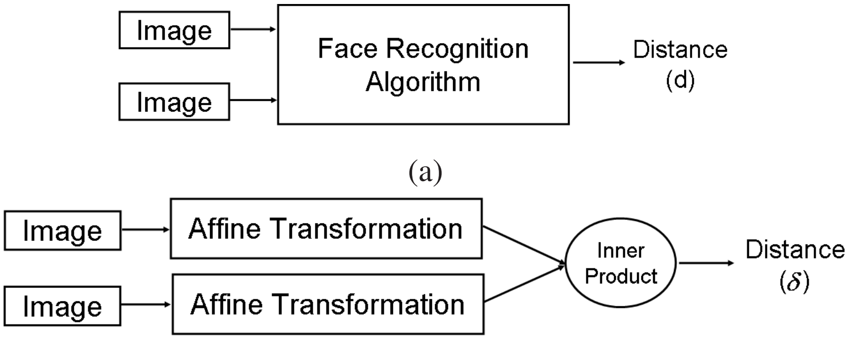
Abstract

We present a theory for constructing linear subspace approximations to face recognition algorithms and empirically demonstrate that a surprisingly diverse set of face recognition approaches can be approximated well using a linear model. A linear model, built using a training set of face images, is specified in terms of a linear subspace spanned by, possibly non-orthogonal, vectors. We divide the linear transformation used to project face images into this linear subspace into two parts: a rigid transformation obtained through principal component analysis, followed by a non-rigid, affine, transformation. The construction of the affine subspace involves embedding of a training set of face images constrained by the distances between them, as computed by the face recognition algorithm being approximated. We accomplish this embedding by iterative majorization, initialized by classical multi-dimensional scaling (MDS). Any new face image is projected into this embedded space using an affine transformation. We empirically demonstrate the adequacy of the linear model using six different face recognition algorithms, spanning both template based and feature based approaches, with a complete separation of the training and test sets. A subset of the Face Recognition Grand Challenge (FRGC) training set is used to model the algorithms and the performance of the proposed modeling scheme is evaluated on the Facial Recognition Technology (FERET) data set. The experimental results show that the average error in modeling for six algorithms is 6.3% at 0.001 False Acceptance Rate (FAR), for the FERET fafb probe set which has 1195 subjects, the most among all the FERET experiments. The built subspace approximation not only matches the recognition rate for the original approach, but the local manifold structure, as measured by the similarity of identity of nearest neighbors, is also modeled well. We found, on an average, 87% similarity of local neighborhood. We also demonstrate the usefulness of the linear model for algorithm dependent indexing of face databases and find that it results in more than 20 times reduction in face comparisons for Bayesian, EBGM, and one proprietary algorithm.

II. Introduction

Intensive research has produced an amazingly diverse set of approaches for face recognition (see [1], [2] for excellent reviews). The approaches differ in terms of the features used, distance measures used, need for training, and matching methods. Systematic and regular evaluations such as the FERET (Facial Recognition Technology) [3], [4], the FRGC (Face Recognition Grand Challenge) [5], [6] and Face Recognition Vendor Test [7], [8] have enabled us to identify the top performing approaches. In general, a face recognition algorithm is a module that computes distance (or similarity) between two face images. Just as linear systems theory allows us to characterize a system based on inputs and outputs, we seek to characterize a face recognition algorithm based on the distances (the "outputs") computed between two faces (the "inputs"). Can we model the distances, d_{ij} computed by any given face recognition algorithm, as a function of the given face images, $\mathbf{x_i}$ and $\mathbf{x_j}$? Mathematically, what is the function ϕ such that $(d_{ij} - ||\phi(\mathbf{x_i}) - \phi(\mathbf{x_j})||)^2$ is minimized? In particular, we consider just affine transforms as it is the simplest model. As we shall see in the experimental section, this affine model suffices for a number of face recognition algorithms. This modeling problem is represented in Fig. 1. Essentially, we seek to infer a subspace that approximates the face recognition algorithm. The transformation allows us to embed a new template, not used for training, into this subspace.

Apart from sheer intellectual curiosity, the answer to this question has some practical benefits. First, a subspace approximation would allow us to characterize face recognition algorithms at a deeper level than just comparing recognition rates. For instance, if ϕ is an identity operator, then it would suggest that the underlying face recognition algorithm is essentially performing a rigid rotation and translation to the face representations similar to principal component analysis



(a)

(b)

Fig. 1. Approximating face recognition algorithms by linear models: Distance between two face images observed by (a) Original face recognition (b) Linear model.

(PCA). If ϕ is a linear operator, then it would suggest that the underlying algorithms can be approximated fairly well by a linear transformation (rotation, shear, stretch) of the face representations. Given training samples, the objective is to approximate the subspace induced by a face recognition algorithm from pairwise relation between two given templates. Experimentally, we have demonstrated that the proposed modeling scheme works well for template based algorithms as well as feature based algorithms. As we shall see, in practice, we have found that a linear ϕ is sufficient to approximate a number of face recognition algorithms, even feature based ones. This raises interesting speculations about the essential simplicity of the underlying algorithms.

Second, if a linear approximation can be built, then it can used to reconstruct face templates just from scores. We have demonstrated this ability in [9]. This has serious security and privacy implications.

Third, we can use the linear subspace approximation of face recognition algorithms to build efficient indexing mechanisms for face images. This is particularly important for the identification scenarios where one has to perform one to many matches, especially using a computationally expensive face recognition algorithm. The proposed model based indexing mechanism has several advantages over two pass indexing scheme. In a two pass indexing scheme, a linear projection such as pca is used to select few gallery images followed by the identification of the probe image within the selected gallery images. However, the performance of these type of system is limited by the performance of the linear projection method even in the presence of a high performing recognition algorithm in the second pass. Whereas the use of linear model of the original algorithm ensures the selection of few gallery images that match those computed by the original algorithm. In Section VII, we have experimentally demonstrated the advantage of the proposed model based indexing scheme over PCA-based modeling scheme using two different face recognition algorithm with more than 1000 subjects in the gallery set.

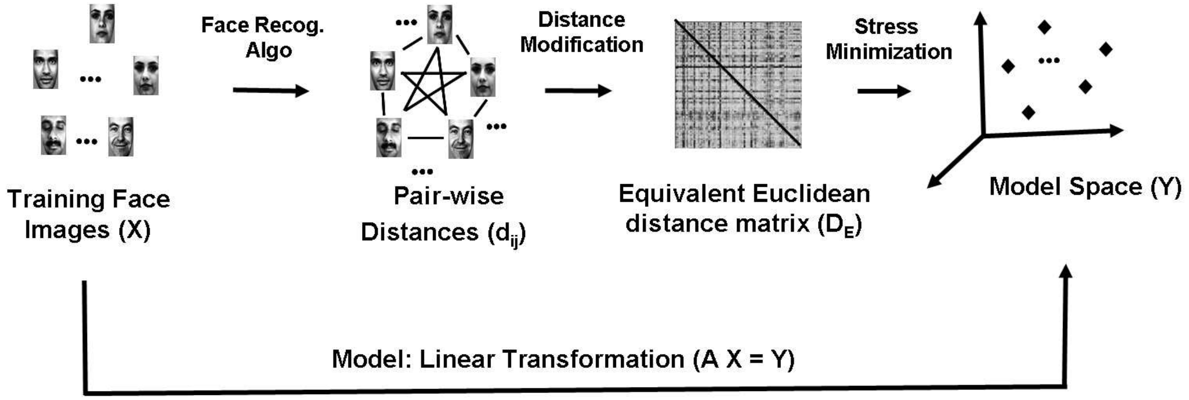

Training Face Images (X)
Face Recog. Algo
Pair-wise Distances (d_{ij})
Distance Modification
Equivalent Euclidean distance matrix (D_E)
Stress Minimization
Model Space (Y)
Model: Linear Transformation (A X = Y)

Fig. 2. Modeling Face Recognition Algorithms: Starting with a given set of training face images, X, we compute the pair-wise dissimilarities $d_{ij}s$ between these images using the underlying face recognition algorithm. We convert the pair-wise dissimilarities to an equivalent Euclidean distance matrices and then use stress minimization method to arrive at the model space. The underlying algorithm is then model by an affine transformation A which transforms the input images X to points of configuration Y in the model space.

We consider ϕ's that are affine transformations, defining a linear subspace spanned by, possibly non-orthogonal, vectors. We treat the algorithm being modeled as a black box. To arrive at this model we need a set of face images (training set) and the distances between these face images, as computed by the face recognition algorithm being approximated. For computational reasons, we decompose the linear model into two parts: a rigid transformation, which can be obtained by any orthogonal subspace approximation of the rigid model such as the principal component analysis (PCA); and a non-rigid, affine transformation. Note that the bases of the overall transformation need not be orthonormal. To construct the affine subspace, we embed the training set of face images constrained by the distances between them, as computed by the face recognition algorithm being modeled. We accomplish this distance preserving embedding by iterative majorization algorithm initialized by classical multi-dimensional scaling (MDS) [10], [11]. This process results in a set of coordinates for the train images. The affine transformation defines the relationship between these embedding coordinates and the rigid (PCA) space coordinates.

We analyze some of the popular face recognition algorithms: eigenfaces (PCA + distance metrics) [12], Linear Discriminant Analysis (LDA) [13], Bayesian Intra/Extra-class person classifier [14], Elastic Bunch Graph Matching (EBGM) [15], Independent Component Analysis (ICA) [16], and one proprietary algorithm. The choice of the face recognition algorithms includes template based approaches such as PCA, LDA, ICA, Bayesian and feature based ones, such as the EGBM and the proprietary algorithm. The Bayesian approach, although template based, actually employs two subspaces to compute the distance, so it is fundamentally different from other linear approaches. One subspace is for inter-subjects variations and the other is for intra-subject variation. We use a subset of the FRGC [5] the training and test the accuracy of the model on the FERET [3] data set for all recognition algorithms, except for the EBGM algorithm. Due to the need for extensive manual intervention in creating ground truth training feature points for the EBGM algorithm, and the non existence of such data for the FRGC data, we use the FERET training set for which ground truth is included in the Colorado State University (CSU) Face Identification Evaluation System [17]. For the proprietary algorithm we use both the FERET training set and a subset of the FRGC training set and compare the modeling results.

The rest of this paper is organized in the following way. In Section III, we review some of the earlier approaches to model recognition performance of different biometric systems as well as the distance based learning approaches using the multi-dimensional scaling approach. In Section IV, we present our approach to model face recognition systems based on match scores. Experimental setup, data sets, and a brief description of different face recognition algorithms used in our experiments are described in Section V. Results of the proposed modeling scheme and indexing of face databases are presented in Section VI and Section VII, respectively. We conclude our work in Section VIII with a summary and discussion of possible extension of the modeling scheme. Note that this training set is the one used to construct the linear model. However, each face recognition algorithm has its own training set that is different from that used for the linear transformation. We have provided specific details in the results section.

III. Related Work

As far as we know, there is no related work that considers the face recognition algorithm modeling problem as we have posed it. This is the first work that seeks to construct a linear transformation to model recognition algorithms. Using the linear model, we also present the first algorithm specific indexing mechanism for face templates and experimentally demonstrate a 20 times reduction in template comparisons on FERET gallery set for the identification scenario.

Perhaps the closest works are those that use multidimensional scaling (MDS) to derive models for standard classifiers such as nearest neighborhood, linear discriminant analysis, and linear programming problem from the dissimilarity scores between objects [18]. A similar framework is also suggested by Roth *et al.* [19], where pairwise distance information is embedded in the Euclidean space, and an equivalence is drawn between several clustering approaches with similar distance based learning approaches. There are also studies that statistically model similarity scores so as to predict the performance of the algorithm on large data sets based on results on small data sets [20]–[23]. For instance, Grother and Phillips [24] proposed a joint density function to independently predict match scores and non-match scores from a set of match scores. Apart from face recognition, methods have been proposed to model and predict performances for other biometric modalities and objects recognition [25], [26].

A couple of philosophical distinctions exists between our work and these related works. First, unlike these works, which try to statistically model the scores, we estimate an analytical model that characterizes the underlying face subspace induced by the algorithm and builds a linear transformation from the original template to this global manifold. Second, unlike some of these methods, we do not place any restrictions on distribution of scores in the training set such as separation between match score distribution and non-match score distribution. We treat the face recognition algorithm to be modeled as a complete black box. Third, we empirically demonstrate the quality of the model under a very strict experimental framework with a complete separation of not only train and test, but also the separation of train sets for the underlying algorithms and that the training set used to build the model.

Perhaps a few words about our previous study [9] is in order. In that work, we briefly introduced the linear modeling scheme and showed that given such a model, we can use it to reconstruct face templates from scores. However, the conclusions were contingent on the ability to construct this linear model. This was demonstrated only for three different face recognition algorithms. In the current work, we have focused on the modeling part. We now have a more sophisticated, two-fold method for building linear models than the single pass approach adopted in [9]. We use iterative stress minimization using a majorization to minimize the error between algorithmic distance and model distance. The output of the classical multi-dimensional scaling initializes this iterative process. The two-fold methods help us in building better generalizable models. The models are better even if the training set used for the face recognition and that used to learn the linear models are different. The empirical conclusions are also based on a more extensive study of six different recognition algorithms. The application to indexing is new as well.

IV. Modeling Face Recognition Algorithm

To model an algorithm from a distance matrix, we need to learn the underlying distribution of face images, the subspace induced by that specific algorithm. We also need a transformation to project new face images into the learned manifold. In the following subsections, we present the mathematical derivation of the proposed affine transformation based modeling scheme for this subspace. Given a set of face images and the pairwise distances between these images, first we compute a point configuration preserving these pairwise distances between projected points on the low dimensional subspace. We use stress minimization with iterative majorization to arrive at a point configuration from match scores between templates on the training set. The iterative majorization algorithm is guaranteed to converge to an optimal point configuration, or in some cases, settles down to a point configuration contributing to a local maxima [11]. However, in either case, an informative initial guess will reduce the number of iterations and speed up the process. We use classical multidimensional scaling for this purpose.

Notations and Definitions: A few notational issues are in order. Let d_{ij} be the dissimilarity between two images, $\mathbf{x_i}$ and $\mathbf{x_j}$ (row-scanned vector representations), ($\mathbf{x_i^T} \in \Re^N$) as computed by the given face recognition algorithm. Here we assume that the face recognition algorithm outputs the dissimilarity scores of two images. However, if a recognition algorithm computes similarities instead of dissimilarities, we can convert the similarity scores s_{ij} into dissimilarities using a variety of transformations, such as $(1-s_{ij})$, $-\log(s_{ij})$, $\frac{1}{s_{ij}}-1$ etc. These distances can then be arranged as a $K \times K$ matrix $\mathbf{D} = [d_{ij}^2]$, where K is the number of images in the training set. In this paper, we will denote matrices by bold capital letters, $\mathbf{A}$, column vectors by bold small letters, $\mathbf{a}$. We will denote the identity matrix by $\mathbf{I}$, a vector of ones by $\mathbf{1}$, a vector of zeros by $\mathbf{0}$, and the transpose of $\mathbf{A}$ by $\mathbf{A^T}$.

We start by considering the difference among a distance metric, an Euclidean distance metric and dissimilarity measure. A dissimilarity (distance) measure $\mathbf{d}$ is a function or association of two objects from one set to a real number. Mathematically, $\mathbf{d} : X \times X \rightarrow \Re$. A smaller value of $\mathbf{d}$ indicates a stronger similarity between two objects and a higher value indicates the opposite. A similarity measure can be considered as the inverse function of dissimilarity measure.

Definition *(Metric Property)* A dissimilarity measure $\mathbf{d} : X \times X \rightarrow \Re$ is called a distance metric if it satisfies the following properties.

1) $d(x,y) = 0$ iff $x = y$ (Reflexive)
2) $d(x,y) \geq 0 \quad \forall x \neq y$ (Positivity)
3) $d(x,y) = d(y,x)$ (Symmetry)
4) $d(x,y) \leq d(x,z) + d(z,y)$ (Triangle inequality)

Note that, a dissimilarity measure may not be a distance metric. However, in applications such as biometrics, the reflexive and positivity property are straight forward. The positivity property can be imparted with a simple translation of dissimilarities values to a positive range. If the distance matrix D violates the symmetric property then we reinstate this property by replacing D with $\frac{1}{2}(D + D^T)$. Although this simple solution will change the performance of the algorithm, this correction can be viewed as a first cut fix for our modeling transformation to the algorithms that violates symmetric property of match scores. In case the dissimilarity measure does not satisfy the symmetry and triangle inequality property, if required, these properties can be imparted if we have set of pair-wise distances arranged in complete distance matrix [9], [10].

Any given dissimilarity matrix $\mathbf{D}$ may violate the metric property and may not be an Euclidean matrix, i.e. a matrix of distances that violates the triangle inequality. However, if $\mathbf{D}$ is not an Euclidean distance matrix then it is possible to derive an equivalent Euclidean distance matrix $\mathbf{D_E}$ from $\mathbf{D}$. We will discuss this in Section IV-B.

A. Computing Point Configuration

The objective is to find a point configuration $\mathbf{Y} = [\mathbf{y_1}, \mathbf{y_2}, \cdots, \mathbf{y_K}]$ such that the squared error in distances $\sum(d_{ij} - \delta_{ij})^2$ is minimum, where d_{ij} is the distance computed between face template $\mathbf{x_i}$ and $\mathbf{x_j}$ and δ_{ij} is the Euclidean distance between configuration points $\mathbf{y_i}$ and $\mathbf{y_j}$. Thus, the objective function can be written as

$$\min \quad \sum_{i \leq j} w_{ij}(d_{ij} - \delta_{ij})^2 \tag{1}$$

where, w_{ij} are weights. The incorporation of weights w_{ij} in Eq. 1 is a generalization of the objective function and can be associated with the confidence in the dissimilarities d_{ij}. For missing values of d_{ij} the corresponding weights w_{ij} is set to zero. In our experiments, the weights are all equal and set to 1. However, for generality, we develop the theory based on weighted scores. Let,

$$\begin{aligned}
\mathbf{S}(\mathbf{Y}) &= \sum_{i<j} w_{ij}(d_{ij}-\delta_{ij})^2 \\
&= \sum_{i<j} w_{ij}d_{ij}^2 + \sum_{i<j} w_{ij}\delta_{ij}^2 - 2\sum_{i<j} w_{ij}d_{ij}\delta_{ij} \\
&= \eta_d^2 + \eta^2(\mathbf{Y}) - 2\rho(\mathbf{Y}) \qquad (2)
\end{aligned}$$

where, η_d^2 is independent of the point configuration $\mathbf{Y}$ and

$$\begin{aligned}
\eta^2(\mathbf{Y}) &= \sum_{i<j} w_{ij}(y_i-y_j)^T(y_i-y_j) \\
&= tr(\mathbf{YVY^T}) \qquad (3)
\end{aligned}$$

where, $v_{ij} = -w_{ij}$ for $i \neq j$ and $v_{ii} = \sum_{j=1,j\neq i}^{n} w_{ij}$.

Similarly,

$$\begin{aligned}
\rho(\mathbf{Y}) &= \sum_{i<j} \frac{w_{ij}d_{ij}}{\delta_{ij}}\delta_{ij}^2 \\
&= \sum_{i<j} \frac{w_{ij}d_{ij}}{\delta_{ij}}(y_i-y_j)^T(y_i-y_j) \\
&= \mathrm{Tr}(\mathbf{YU(Y)Y^T}) \qquad (4)
\end{aligned}$$

where, $[\mathbf{U}(\mathbf{Y})] = [u_{ij}]$ and

$$\begin{aligned}
u_{ij} &= \begin{cases} \frac{-w_{ij}d_{ij}}{\delta_{ij}} & \text{if } \delta_{ij} \neq 0 \text{ and } i \neq j \\ 0 \quad \text{otherwise} \end{cases} \\
u_{ii} &= \sum_{j=1,j\neq i} u_{ij}
\end{aligned}$$

Hence, from Eqn. 2,

$$\mathbf{S}(\mathbf{Y}) = \eta_d^2 + \mathrm{Tr}(\mathbf{YVY^T}) - 2\mathrm{Tr}(\mathbf{YU(Y)Y^T}) \qquad (5)$$

The configuration points Y can be found by maximizing $\mathbf{S}(\mathbf{Y})$ in many different ways. In this work, we consider the *iterative majorization* algorithm proposed by Borg and Groenen [11]. Let,

$$\mathbf{T}(\mathbf{Y},\mathbf{Z}) = \eta_d^2 + \mathrm{Tr}(\mathbf{TVY^T}) - 2\mathrm{Tr}(\mathbf{YU}(\mathbf{Z})\mathbf{Z^T}) \tag{6}$$

then $\mathbf{T} \geq \mathbf{S}$ and $\mathbf{T}(\mathbf{Y},\mathbf{Y}) = \mathbf{S}(\mathbf{Y})$, hence $\mathbf{T}(\mathbf{Y},\mathbf{Z})$ majorizes $\mathbf{S}(\mathbf{Y})$. So the optimal set of configuration points $\mathbf{Y}$ can be found as follows

$$\begin{aligned} \frac{\delta \mathbf{T}}{\delta \mathbf{Z}} &= 0 \\ 2\mathbf{VY} - \mathbf{2U}(\mathbf{Z})\mathbf{Z^T} &= 0 \end{aligned} \tag{7}$$

Thus, the iterative formula to arrive at the optimal configuration points can be written as follows

$$\mathbf{Y}^{(k)} = \mathbf{V}^{\dagger}\mathbf{U}(\mathbf{Y}^{(k-1)})(\mathbf{Y}^{(k-1)})^T \tag{8}$$

where, $\mathbf{Y}^{(0)}$ is the initial configuration points and $\mathbf{V}^{\dagger}$ represent the pseudo-inverse of $\mathbf{V}$.

B. Choice of initial point configuration

Although the iterative solution presented in Eqn. 8 can be initialized with any random starting configuration point, an appropriate guess will reduce the number of iteration to find optimal configuration points. We initialize the iterative algorithm with a set of configuration points derived by applying classical multi-dimensional scaling on original distance matrix. Classical multidimensional scaling works well when the distance measure is a metric or more specifically an Euclidean distance matrix. Therefore, we first compute an approximate Euclidean distance $\mathbf{D_E}$ from the original distance matrix $\mathbf{D}$ followed by the derivation of initial configuration points using classical multidimensional scaling adapted from Cox and Cox [10].

Computing Equivalent Euclidean distance matrix ($\mathbf{D_E}$)

Given the original distance matrix $\mathbf{D}$, we first check if the distance matrix satisfies the Euclidean distance properties. If any such property is violated, then we replace the original distance matrix $\mathbf{D}$ with an equivalent distance matrix D_E. The term "equivalent" is used in the sense that, the overall objective of both the distance matrix $\mathbf{D}$ and $\mathbf{D_E}$ remains same. For example, in our case, adding a constant to all the entries of original distance matrix $\mathbf{D_E}$ does not alter the

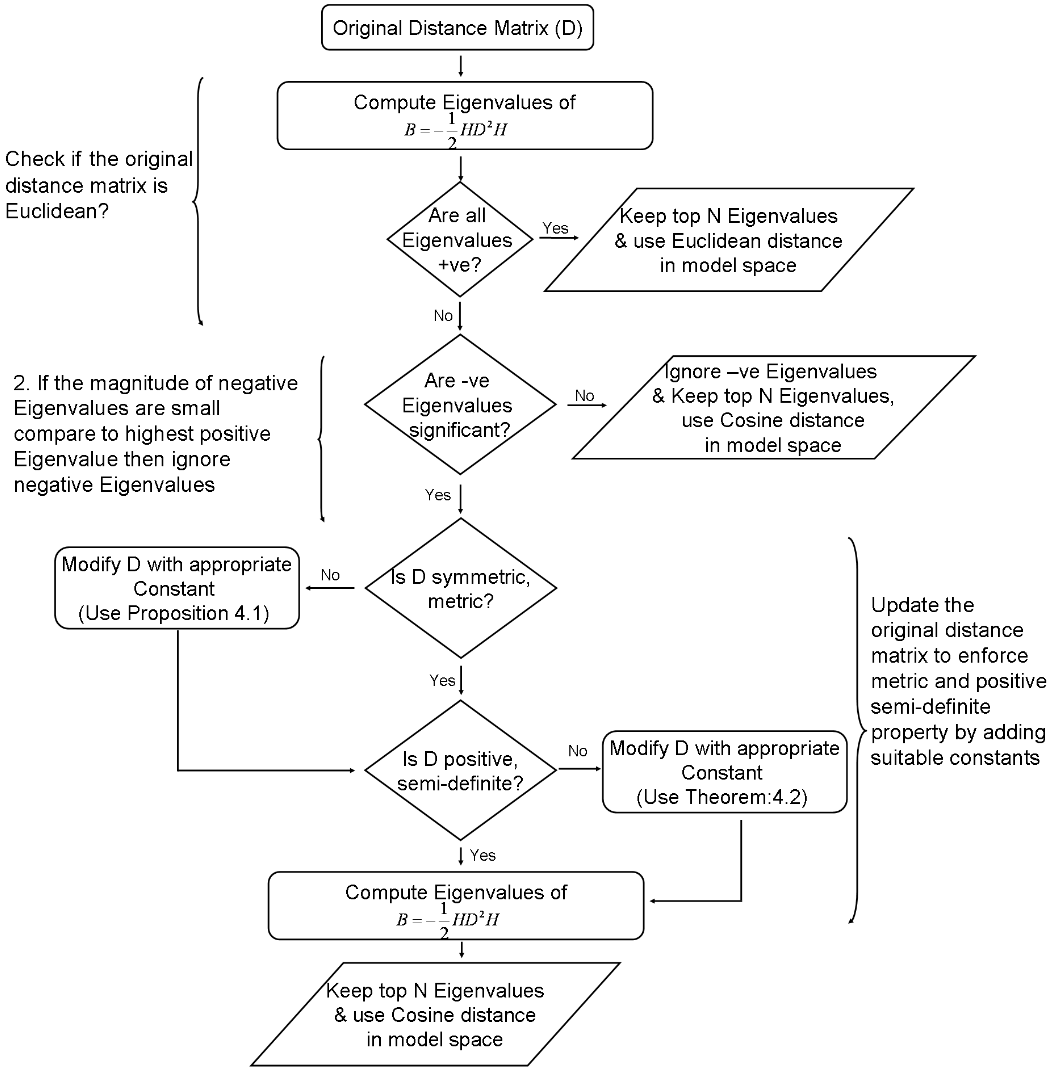

Original Distance Matrix (D)
Compute Eigenvalues of
$B=-\frac{1}{2}HD^2H$
Check if the original distance matrix is Euclidean?
Are all Eigenvalues +ve?
Yes
Keep top N Eigenvalues & use Euclidean distance in model space
No
Are -ve Eigenvalues significant?
No
Ignore –ve Eigenvalues & Keep top N Eigenvalues, use Cosine distance in model space
2. If the magnitude of negative Eigenvalues are small compare to highest positive Eigenvalue then ignore negative Eigenvalues
Yes
Is D symmetric, metric?
No
Modify D with appropriate Constant (Use Proposition 4.1)
Yes
Is D positive, semi-definite?
No
Modify D with appropriate Constant (Use Theorem:4.2)
Update the original distance matrix to enforce metric and positive semi-definite property by adding suitable constants
Yes
Compute Eigenvalues of
$B=-\frac{1}{2}HD^2H$
Keep top N Eigenvalues & use Cosine distance in model space

Fig. 3. Steps to compute initial configuration points $Y^{(0)}$. If the given match scores are similarity measures between face images then we convert it dissimilarities (D). We verify the Euclidean property of the distance matrix D and compute an equivalent Euclidean distance matrix D_E, if necessary. The dimension of the model space is also determined during the process.

overall performance of a face recognition system and hence, have similar behavior in terms of recognition performances.

If the original distance matrix **D** is not Euclidean, as in case of most of the face recognition algorithms, then we use the following propositions to derive an equivalent Euclidean distance matrix $\mathbf{D_E}$ from **D**. Given an arbitrary matrix **D**, we enforce the metric property using Prop. 4.1 then we convert the metric (distance) matrix to an Euclidean distance matrix using Thm 4.2.

Proposition 4.1: If $\mathbf{D}$ is non-metric then the matrix $[d_{rs}+c],(r \neq s)$ is metric where $c \geq \max_{i,j,k}|d_{ij}+d_{ik}-d_{jk}|$ [10], [27].

Theorem 4.2: If $\mathbf{D}=[d_{ij}^2]$ is a metric distance, then there exists a constant h such that the matrix with elements $(d_{ij}+h)^{\frac{1}{2}}, i \neq j$ is Euclidean, where $h \geq -\lambda_n$ is the smallest (negative) eigenvalue of $\frac{1}{2}\mathbf{HDH}$, where $\mathbf{H}=(\mathbf{I}-\frac{1}{K}\mathbf{1}\mathbf{1}^T)$ [10], [27].

In Fig. 3, we outline the steps involved to modify the original distance and determining the dimension of the model space. The dimension of the model space is determined by computing the eigenvalues of the matrix $\frac{1}{2}\mathbf{HDH}$ defined in Theorem 4.2. The eigen-spectrum of the matrix $\frac{1}{2}\mathbf{HDH}$ provides an approximation to the dimension of the projected space. The dimension of the model space is decided in a more conventional way of neglecting smaller eigenvalues and keeping 99% of the energy of eigen-spectrum of $\mathbf{B}$. In the presence of negative eigenvalues with high magnitude, Pekalaska and Duin [28] suggested a new embedding scheme of the data points in pseudo-Euclidean space whose dimension is decided by both positive and negative eigenvalues of high magnitudes. However, since in our case, we have modified the original distance matrix to enforce the Euclidean property, we do not have large magnitude negative eigenvalues of the modified distance matrix.

The flow chart in Fig. 3 is divided into three important blocks, demarcated by curly braces with comments. In the first block, the original dissimilarity matrix $\mathbf{D}$ or a similarity matrix converted to dissimilarity matrix with a suitable function is tested for the Euclidean property. If $\mathbf{D}$ is Euclidean then classical multidimensional scaling at the very first iteration will result the best configuration points $\mathbf{y_i}$. The subsequent iterative process using stress minimization will not result in any improvement so remaining steps can be skipped. Furthermore, we would infer that the face recognition algorithm uses Euclidean distance as the distance measure. So in this particular case, we can use the Euclidean distance measure in the model space as well.

If the original dissimilarity matrix $\mathbf{D}$ is not Euclidean then in next two blocks, we find those properties of Euclidean distance matrix that are violated by $\mathbf{D}$ and reinforce those properties by deriving approximated Euclidean distance matrix $\mathbf{D_E}$ from $\mathbf{D}$. Henceforth, we use classical MDS on $\mathbf{D_E}$ to determine the dimension of the model space as well to arrive at an initial set of configuration points $\mathbf{Y}^{(0)}$. At this point, we do not have any knowledge about the distance measure used by the original algorithm but we know that original distance matrix is not Euclidean, so we consistently use cosine distance measure for such model spaces.

C. Classical Multidimensional Scaling

Given the equivalent Euclidean distance matrix $\mathbf{D_E}$, here the objective is to find K vectors, $\{\mathbf{y_1}, \cdots, \mathbf{y_K}\}$ such that

$$\mathbf{D_E}(i,j) = (\mathbf{y_i} - \mathbf{y_j})^T(\mathbf{y_i} - \mathbf{y_j}) \tag{9}$$

Eq. 9 can be compactly represented in matrix form as

$$\mathbf{D_E} = \mathbf{c} \cdot \mathbf{1}^T + \mathbf{1} \cdot \mathbf{c}^T - 2\mathbf{Y}^T\mathbf{Y} \tag{10}$$

where $\mathbf{Y}$ is matrix constructed using the vectors $\mathbf{y_i}$ as the columns $\mathbf{Y} = [\mathbf{y_1}, \cdots, \mathbf{y_K}]$ and $\mathbf{c}$ is a column vector of the magnitudes of the vectors $\mathbf{y_i}$'s. Thus

$$\mathbf{c} = [\mathbf{y_1}^T\mathbf{y_1}, \cdots, \mathbf{y_K}^T\mathbf{y_K}]^T \tag{11}$$

Note that the above configuration points $\mathbf{y_i}$'s are not unique. Any translation or rotation of vectors $\mathbf{y_i}$'s can also be a solution to Eq. 9. To reduce such degrees of freedom of the solution set, we constrain the solution set of vectors to be centered at the origin and the sum of the vectors to zero, i.e. $\sum_i \mathbf{y_i} = \mathbf{0}$.

To simplify Eq. 10, if we pre- and post-multiple each side of the equation by centering matrix $\mathbf{H} = (\mathbf{I} - \frac{1}{K}\mathbf{1}\mathbf{1}^T)$, we have

$$\mathbf{B} = -\frac{1}{2}\mathbf{H}\mathbf{D_E}\mathbf{H} = \mathbf{Y}^T\mathbf{Y} \tag{12}$$

Since $\mathbf{D}_E$ is Euclidean matrix, the matrix $\mathbf{B}$ is represents the inner product between the vectors, $\mathbf{y_i}$, and is a symmetric, positive semi-definite matrix [10], [11]. Solving Eq. 12 yields the initial configuration points as

$$\mathbf{Y^{(0)}} = (\mathbf{V_{EVD}^M} \Delta_{\mathbf{EVD}}^{M}{}^{\frac{1}{2}})^T \tag{13}$$

where $\Delta_{\mathbf{EVD}}^M$ is a $M \times M$ diagonal matrix consisting of M non-zero eigenvalues of $\mathbf{B}$ and $\mathbf{V_{EVD}^M}$ represents the corresponding eigenvectors of $\mathbf{B}$.

D. Solving for Base Vectors

So far, we have seen how to find a set of coordinates, $\mathbf{Y^{(n)}}$, such that the Euclidean distance between these coordinates is related to the distances computed by the recognition algorithm by an additive constant. We now find an affine transformation, $\mathbf{A}$, that will relate these coordinates, $\mathbf{Y}$, to the images, $\mathbf{X}$, such that

$$\mathbf{Y^{(n)}} = \mathbf{A}(\mathbf{X} - \mu) \tag{14}$$

where μ is the mean of the images in training set, i.e. average face. We do not restrict this transformation to be orthonormal or rigid. We consider $\mathbf{A}$ to be composed of two sub-transformations: non-rigid transformation $\mathbf{A_{nr}}$ and rigid transformation rigid, $\mathbf{A_r}$, i.e., $\mathbf{A} = \mathbf{A_{nr}}\mathbf{A_r}$. The rigid part $\mathbf{A_r}$ can be arrived at by any analysis that computes an orthonormal subspace from the given set of training images. In this experiment, we use the principal component analysis (PCA) for the rigid transformation. Let the PCA coordinates corresponding to the non-zero eigenvalues, i.e. non-null subspace, be denoted by $\mathbf{X_r} = \mathbf{A_r}(\mathbf{X} - \mu)$. The non-rigid transformation, $\mathbf{A_{nr}}$, relates these rigid coordinates, $\mathbf{X_r}$ to the distance based coordinates, $\mathbf{Y}$. From Eq. 14

$$\mathbf{Y^{(n)}} = \mathbf{A_{nr}}\mathbf{X_r} \tag{15}$$

Multiplying both sides of Eq. 15 by $\mathbf{X_r}^T$ and using the result that $\mathbf{X_r}\mathbf{X_r}^T = \Lambda_{PCA}$, where Λ_{PCA} is the diagonal matrix with the non-zero eigenvalues computed by PCA, we have

$$\mathbf{A_{nr}} = \mathbf{Y^{(n)}}\mathbf{X_r}^T\Lambda_{\mathbf{PCA}}{}^{-1} \tag{16}$$

This non-rigid transformation allows for shear and stress, and the rigid transformation, computed by principal component analysis, together model the face recognition algorithm. Note that the rigid transformation is not dependent on the face recognition algorithm; it is only the non-rigid part that is determined by the distances computed by the recognition algorithm. An alternative viewpoint could be that the non-rigid transformation captures the difference between a PCA based recognition strategy – the baseline – and the given face recognition algorithm.

Thus the overall outline of modeling approach can be summarized as follows.

- **Input:**
  1) A training set containing K face images
  2) The dissimilarity/Similarity matrix 'D' computed on the training using the face recognition algorithm
- **Algorithm:**
  1) Check if $\mathbf{D}$ is Euclidean, if necessary convert to an equivalent Euclidean distance matrix $\mathbf{D_E}$.
  2) Compute initial configuration points $Y^{(0)}$ (See Fig. 3)
  3) Use the iterative scheme in Eqn. 8 to arrive at the final configuration points. The iteration is terminated when the error $S(Y)$ is less than the tolerance parameter ε which is empirically set to 0.001 in our experiments.

4) Compute the rigid sub-transformation A_r using Principal Component Analysis on training set.
5) Compute the non-rigid sub-transformation A_{nr}, as shown in Eqn. 16
6) $A = A_r A_{nr}$ is the required model affine transformation

V. Experimental Setup

We evaluate the accuracy of the proposed linear modeling scheme using six fundamentally different face recognition algorithms and compare recognition performances of each of these algorithms with corresponding models. We demonstrate the consistency of modeling scheme on FERET face data sets. In the following subsections, we provide more details about the face recognition algorithms and the distance measures associated with these algorithms, train and test sets used in our experiments, and the metrics used to evaluate the strength of the purposed modeling scheme.

Experimental results, presented in the next section, validate that the proposed linear modeling scheme generalizes across probe sets representing different variations in face images (FERET probe sets). We also demonstrate that different distance measures coupled with the PCA algorithm, and normalization of match scores (discussed in Section VI-C), have minimal impact on the proposed modeling approach. In Section VII, we also demonstrate the usefulness of such modeling schemes towards algorithm dependent indexing of face databases. The indexing of face images using the proposed modeling scheme substantially reduce the computational burden of the face recognition system in the identification scenarios.

A. Data Sets

The FERET data set [3], used in this experiment, is publicly available and equipped with predefined training, gallery, and probe sets commonly used to evaluate face recognition algorithms. The FERET data set contains 1196 number of distinct subjects in the gallery set. We use a subset of FRGC [6] training set containing 600 images from the first 150 subjects (increasing order of the id) to train our model. This data set was collected at a later date, at a different site, and with different subjects than FERET. Thus, we have a strong separation of train and test set.

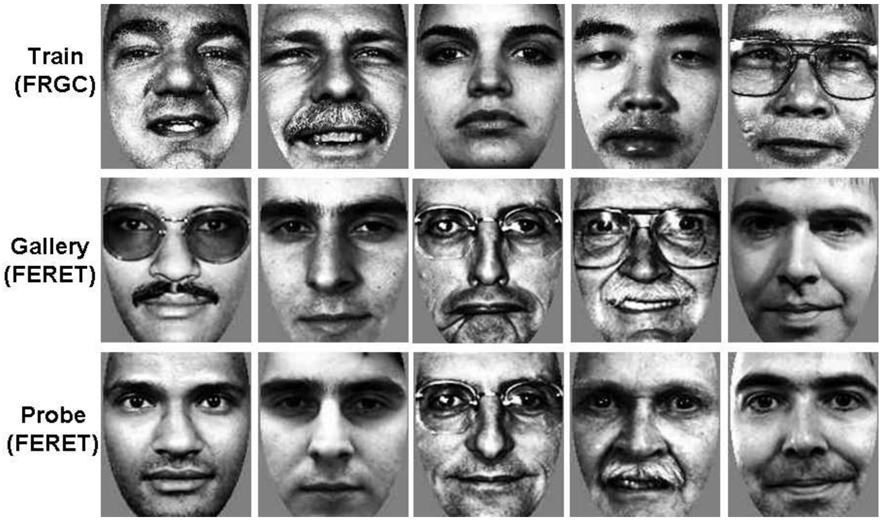


Fig. 4. Sample face images: Top row represents sample training images from the FRGC training set. Middle row represents sample gallery images from the FERET data set and bottom row represents sample probe images with different variations.

B. Face Recognition Algorithms & Distance Transformation

We evaluate our proposed modeling scheme with four different template based algorithms and two feature based face recognition algorithms. The template based approaches include Principal Component Analysis (PCA) [12], Independent Component Analysis (ICA) [16], Linear Discriminant Analysis (LDA) [13], and Bayesian Intrapersonal/Extrapersonal Classifier (BAY). Note that the Bayesian (BAY) algorithm employs two subspaces to compute the distance. A proprietary algorithm (PRP) and Elastic Bunch Graph Matching (EBGM) [15] algorithm are selected to represent the feature based recognition algorithms. For further details on these algorithms, the readers may refer to the original papers or recent surveys on face recognition algorithms [1], [2]. All the face images used in this experiment, except for EBGM algorithm, were normalized geometrically using the Colorado State University (CSU) Face Identification Evaluation System [17] to have the same eye location, the same size (150 x 130) and similar intensity distribution. Few normalized face images are shown in Fig. 4. The EBGM algorithm requires a special normalization process for face images that is manually very intensive. So, we use the training set that is provided with the CSU data set [17] to train the model for the EBGM algorithm. This training set is part of the FERET data set, but different from the probe set used in the experiments.

TABLE I

SIMILARITY/DISSIMILARITY MEASURES OF DIFFERENT FACE RECOGNITION ALGORITHMS

Algorithm	Measure	Similarity/Dissimilarity	Range of Scores	Transformation
PCA	Cosine distance in Mahalanobis Space	Dissimilarity	-1 to 1	$1+d_{ij}$
LDA	Euclidean	Dissimilarity	0 to ∞	none
ICA	Cosine angle in Mahalanobis Space	Dissimilarity	-1 to 1	$1+d_{ij}$
BAY	Probability of Intra/Extra Class of the template	Dissimilarity	0 to ∞	none
EBGM	Face Graph Narrowing Local Search	Dissimilarity	-1 to 0	$-log(-d_{ij})$
PRP	Unknown	Similarity	a to b	$b-d_{ij}$

The six face recognition algorithms and the distance measures associated with each of these algorithms are summarized in Table I. Except for the proprietary and the ICA algorithms, the implementation of all other algorithms are publicly available at Colorado State University (CSU) Face Identification Evaluation System [17]. The implementation of the ICA algorithm has been adapted from [29]. The particular distance measures for each of these algorithms are selected due to their higher recognition rates compared to other possible choice of distance measures. The last two columns in Table I indicate the range of the similarity/dismiliarity scores of the corresponding algorithms and the transformation used to convert these scores to a range such that the lower range of all the transformed distances are the same (i.e. the distance between two similar face images are close to 0). The distance measure for the Bayesian Intrapersonal/Extrapersonal Classifier is a probability measure but due to the numerical challenges associated with small probability values, the distances are computed as the approximations to such probabilities. The implementation details of distance measures for the Bayesian algorithm and the EBGM algorithm can be found in [30] and [31] respectively. Also, in addition to the above transformations, the distance between two exact image is set to zero in order to to maintain the reflexive property of the distance measure. All the above mentioned distance measures also exhibit symmetric property; thus, no further transformation is required to enforce the symmetric property of the distance measure.

C. Train and Test Sets

Out of six selected algorithm, except for the proprietary algorithm, the other five algorithms require a set of face images for the algorithm training process. This training set is different from the training set required to model the individual algorithms. Therefore, we define two training sets; an algorithm train set (*algo-train*) and a model train set (*model-train*). We use a set of 600 controlled images from 150 subjects (in the decreasing order of their numeric id) from the FRGC training set to train the individual algorithms (*algo-train*). To build the linear model for each algorithm, we use another subset of the FRGC training set with 600 controlled images from the first 150 subjects (in the increasing order of their numeric id) with four images per subjects (*model-train*). Due to limited number of subjects in the FRGC training set, few subjects appear in both the training set; however, there is no common image in *algo-train* and *model-train* set. The feature based EBGM algorithm differs from other algorithms with a special normalization and localization process of face images and requires manual landmark points on training images. This process is susceptible to errors and need to be done carefully. So, instead of creating our own ground truth features on a new data set for the EBGM algorithm, we use the FERET training set containing 493 images provided in CSU face evaluation system including the special normalized images required for EBGM algorithm. Since this training set has been used widely, we have confidence on its quality. The *algo-train* and the *model-train* for EBGM algorithm are same.

The proprietary algorithm does not require any training images. However, while building a model for the proprietary algorithm, we empirically observed that the performance of the linear model demonstrates higher accuracy on the FERET probe sets when the model is trained (*model-train*) on the FERET training set. In the result section, we have demonstrated the performance of our linear model to the proprietary algorithm with these two different *model-train* sets.

To be consistent with other studies, for test sets, we have selected the gallery set and four different probe sets as defined in the FERET data set. The gallery set contains 1196 face images of 1196 subjects with neutral or minimal facial expression and with frontal illumination. Four sets of probe images (fb, fc, dupI, dupII) are created to verify the recognition performance under four different variations of face images. If the model is correct the algorithm and model performances should match for all these probe conditions. The ‘fb’ set contains 1195 images

TABLE II

SUMMARY OF TRAIN AND TEST SETS

Algorithm	algo-train	model-train [a]	Test Sets
PCA	600 images (FRGC train set)	600 images (FRGC train set)	FERET Probe Sets
LDA	600 images (FRGC train set)	600 images (FRGC train set)	FERET Probe Sets
ICA	600 images (FRGC train set)	600 images (FRGC train set)	FERET Probe Sets
BAY	600 images (FRGC train set)	600 images (FRGC train set)	FERET Probe Sets
EBGM	493 images (FERET train set)	493 images (FERET train set)	FERET Probe Sets
PRP	Unknown	600 images (FRGC train set)	FERET Probe Sets
PRP	Unknown	493 images (FERET train set)	FERET Probe Sets
PRP	Unknown	2064 images (FRGC train set)	FRGC Exp1, Exp2, Exp4

[a] Theere are no common images in the algo-train set and the model-train set for PCA, LDA, ICA, and BAY algorithm (see text)

from 1195 subjects with different facial expression than gallery images. The ‘fc’ set contains 194 images from 194 subjects with different illumination conditions. Both ‘fb’ and ‘fc’ images are captured at the same time as that of gallery images. However, 722 images from 243 subjects in probe set ‘dupI’ are captured between in between 0 to 1031 days after the gallery images were captured. Probe set ‘dupII’ is a subset of probe set ‘dupI’ containing 234 images from 75 subjects which were captured at least one and a half year after the gallery images. The above mentioned numbers of images in probe and gallery sets are pre-defined within the FERET distribution.

D. Performance Measures to Evaluate the Linear Model

We compare the recognition rates of the algorithms with recognition rates of the linear models in terms of standard Receiver Operating Characteristic (ROC). Given the context of biometrics, this is a more appropriate performance measure than the error in individual distances. How close is the performance of the linear model to that of the actual algorithm on image sets that are different from the train set?

In addition to comparison of ROC curves, we use the Error in Modeling measure, to quantify the accuracy of the model at a particular False Acceptance Rate (FAR). We compute the Error in the modeling by comparing the True Positive Rate (TPR) of the linear model with the TPR of the original algorithm at a particular False Positive Rate (FAR).

$$\text{Error in Modeling } (\%) = \frac{abs(TPR_{orig} - TPR_{model})}{max(TPR_{orig}, TPR_{model})} \times 100 \tag{17}$$

where, TPR_{orig} and TPR_{model} are the true positive rate of the original algorithm and true positive rate of the model at a particular FAR.

In order to closely examine the approximating linear manifold, we also define a stronger metric, Nearest Neighbor Agreement, to quantify the local neighborhood similarity of face images in approximating subspace with the original algorithm. For a given probe P_k, let G_i be the nearest subject as computed by the algorithm and G_j be the nearest subject based on the linear model. Let,

$$s_k(G_i, G_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

Then the nearest neighbor agreement between the model and the original algorithms is quantified as

$$S = (\frac{1}{P}\sum_{k=1}^{P} s_k) \times 100$$

where P is the total number of probes in the probe set. Note that the Nearest Neighbor Agreement metric S is a stronger metric than rank 1 identification rate in Cumulative Match Curves (CMCs). Two algorithms can have the same rank 1 identification but the Nearest Neighbor Agreement can be low. For the later to be high, the identities of both the correct and incorrect matches should agree. In other words, a high value of this measure indicates that the model and the original algorithm agree on neighborhood structure of the face manifold.

VI. Modeling Results

In this section, we present experimental results of our proposed linear models to the six different face recognition algorithms using the FERET probe sets. Using the metrics defined in previous section, we demonstrate the strength of the linear model on FERET data set and with complete separation of training and test sets. The experimental results show that the average Error in Modeling for six algorithms is 6.3% for fafb probe set which contains a maximum number of subjects among all the four probe sets. We also observe that the proposed linear model exhibits an average of 87% accuracy when measured for the similar neighborhood relationship with the original algorithm. A detailed analysis and explanation of these results are presented in the following sub-sections.

A. Recognition Performances

In Figs. 5 to 10, we show performance of each for the six face recognition algorithms, respectively. In each figure, we have four plots, corresponding to the four different FERET probe sets. In each of these subplots, we show the ROCs for the original algorithm along with the performance of the linear approximation. We should compare how closely these two ROCs match, in each individual plot. Note the log-scale for the false alarm rate. We observe that not only the recognition performance of the model matches with that of the original algorithm, but it also generalizes to the variations in face images represented by four different probe sets. For example, the performance of ICA algorithm in fafc (Fig. 7(b)) is lower as compared to rest of the algorithm and the modeling performance is also lower for ICA algorithm which is a good indication of an accurate model of the underlying algorithm. Similar performances can also be observed in case of LDA and BAY algorithms. This is evidence of the generalizability of the learnt model across different conditions.

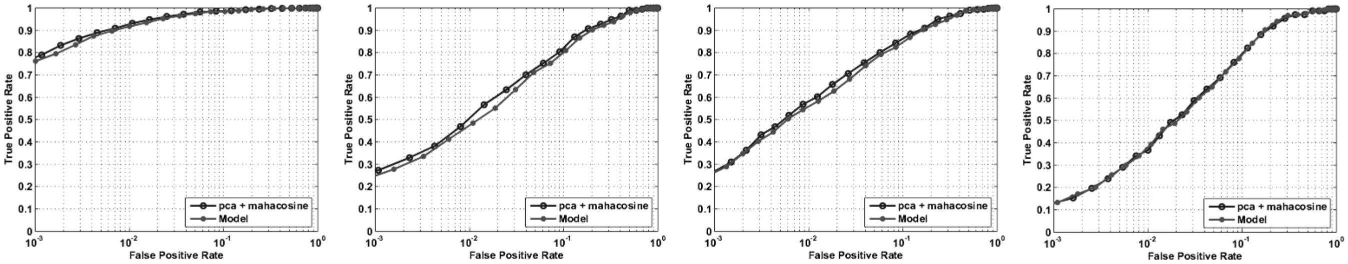


Fig. 5. ROC Curves: Comparison of recognition performance of PCA algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set.

Also, for fafb probe set, the error in the modeling of all the algorithms at 0.001 FAR are 3.8%, 7%, 9%, 5%, 4% and 26% for PCA, LDA, ICA, BAY, EBGM and PRP algorithms, respectively. The high error rate for the PRP algorithm indicates that the linear model for PRP algorithm is under-trained. Note that, the training set used for the proprietary algorithm or the score normalization techniques adapted to optimize the performances are unknown. If we use our linear model for the PRP algorithm with the FERET training set containing 493 images and also study the effect of two standard score normalization methods on the proposed linear model for the proprietary algorithm. The performance of the PRP algorithm on four FERET probe sets

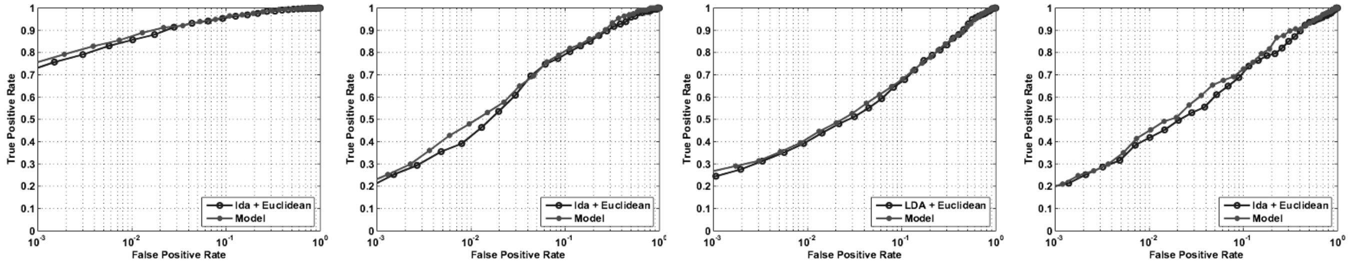


Fig. 6. ROC Curves: Comparison of recognition performance of LDA algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set.

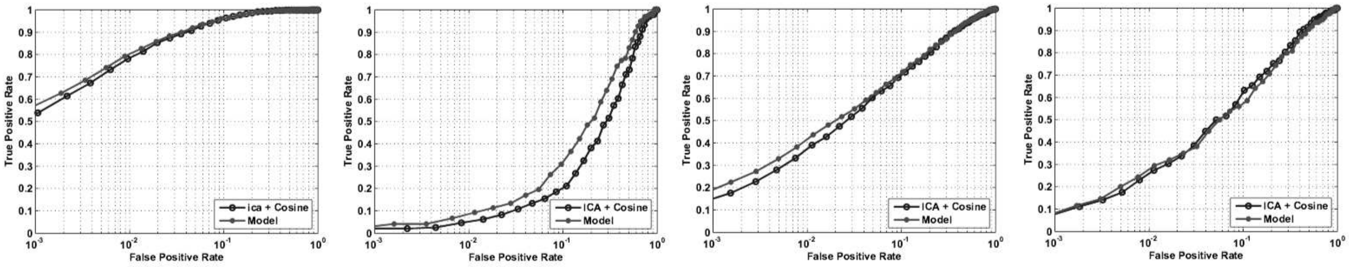


Fig. 7. ROC Curves: Comparison of recognition performance of ICA algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set.

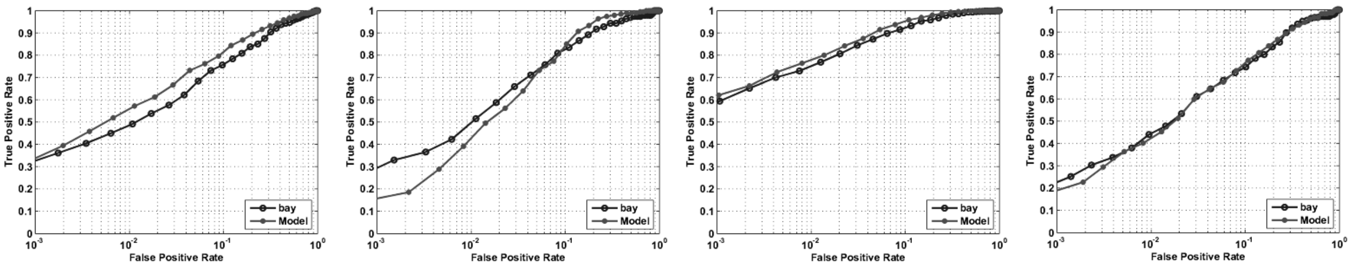


Fig. 8. ROC Curves: Comparison of recognition performance of BAY algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set.

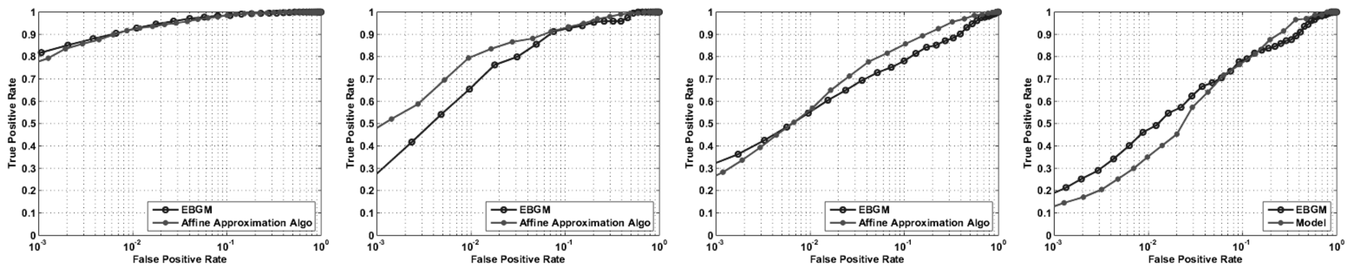


Fig. 9. ROC Curves: Comparison of recognition performance of EBGM algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set.

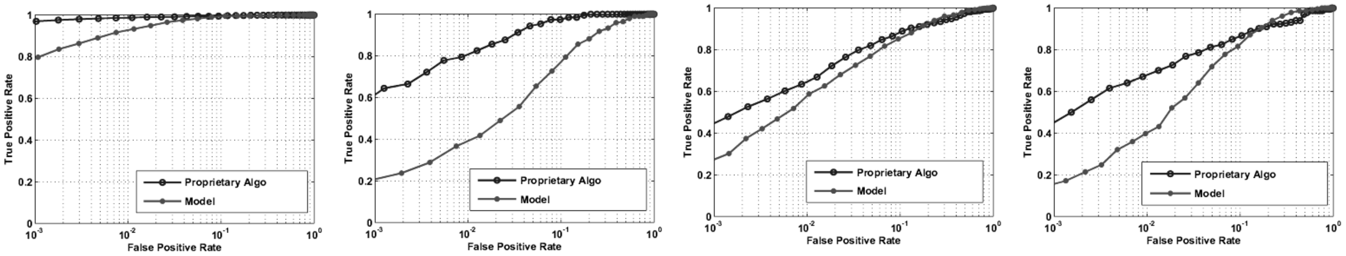


Fig. 10. ROC Curves: Comparison of recognition performance of PRP algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set.

and the performance of the linear model trained using the FERET training set are presented in Fig 11. With the FERET training set, the Error in modeling for PRP algorithm in fa-fb probe set is reduced to 13%, and with the normalization process, the error in modeling for the proprietary algorithm is further reduced to 9%. The effect of score normalization on the proposed modeling scheme is discussed in Section VI-C.

B. Local Manifold Structure

Fig. 12 shows the similarity of the neighborhood relationship for six different algorithms on the FERET fafb probe set. Observe that, irrespective of correct or incorrect match, the Nearest Neighbor Agreement metric has an average accuracy of 87% on all the six algorithms. It is also important to note that, for algorithms where the performance of the model is better than that of the original algorithm, the metric *S* is penalized for such improvement in the performances,

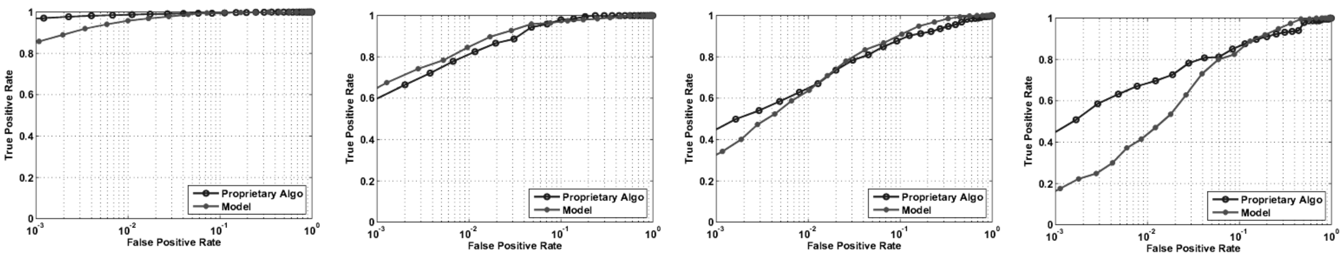


Fig. 11. ROC Curves: Comparison of recognition performance of PRP algorithm with corresponding linear model on (from left to right) FERET-fafb, FERET-fafc, FERET-dupI and FERET-dupII probe set with FERET gallery set. The linear model is trained using 493 FERET training images.

and pulls down the subject agreement values even if the model has better performance than the original algorithm. This is appropriate in our modeling context because the goal is to model the algorithm not necessarily to better it. The high value of such a stringent metric validates the strength of the linear model. Even with little information about the train and optimization process of the proprietary algorithm, the linear model still exhibits a 70% nearest neighborhood accuracy for the proprietary algorithm. As we observe from Fig. 10 and Fig. 11, the proprietary algorithm might have been optimized for FERET type data sets and may have used some score normalization techniques to transform the raw match scores to a fixed interval. In the next subsection, we explore the variation in the model's performance with different distance measures using PCA algorithm as well as the effect of score normalization on our proposed modeling scheme using the proprietary algorithm.

C. Effect of Distance Measures and Score Normalization

Different face recognition algorithms use different distance measures and, in many cases the distance measure is unknown and non-Euclidean in nature. In order to study the effect of various distance measures on the proposed modeling scheme, we use PCA algorithm with six different distance measures as mentioned in the first column of Table III. For a stronger comparison, we kept all other parameters, such as the training set and dimension of the PCA space same. Only the distance measure is changed. These distance measures are implemented in CSU face evaluation tool, and we use them as per the definition in [17]. In Table III, we present the Error

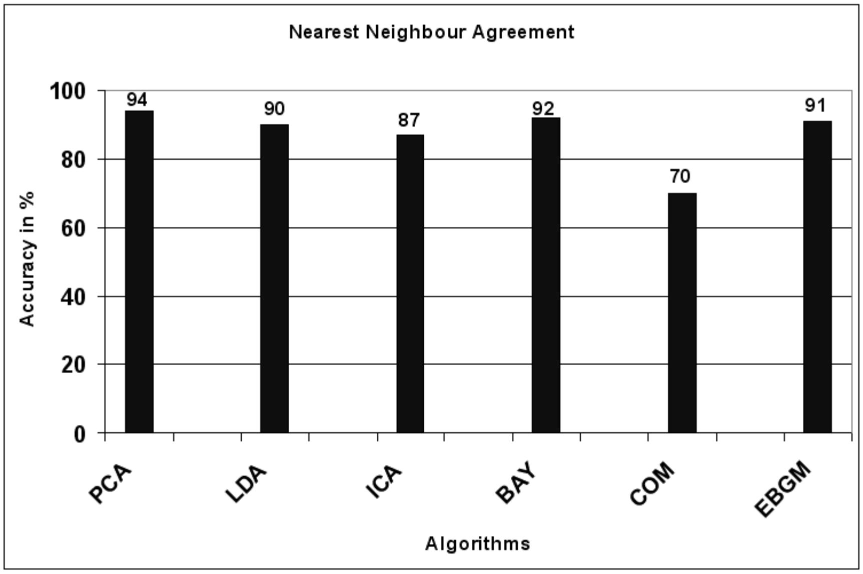


Fig. 12. Similarity of the local manifold structure between the original algorithm and the linear model as captured by the nearest neighbor agreement metric using the FERET fafb probe set. The number of times the algorithm and model agree on subjects irrespective of genuine or imposter match are shown in percentage. Note that, this metric is a stronger measure than the rank-1 identification rate in CMC analysis.

in Modeling (see Eqn. 17) for the PCA algorithm with different distance measures on the FERET fafb probe set. The implementation details of these distance measures are described in [17]. Note that, as described in Fig 3, except for PCA+Euclidean distance, the model uses cosine distance for all other cases. From the table, we observe that for different distance measures, the Error in modeling is in the magnitude of $10^{-2}\%$ or less. Thus, it is apparent that different distance measure have minimal impact on the proposed modeling scheme.

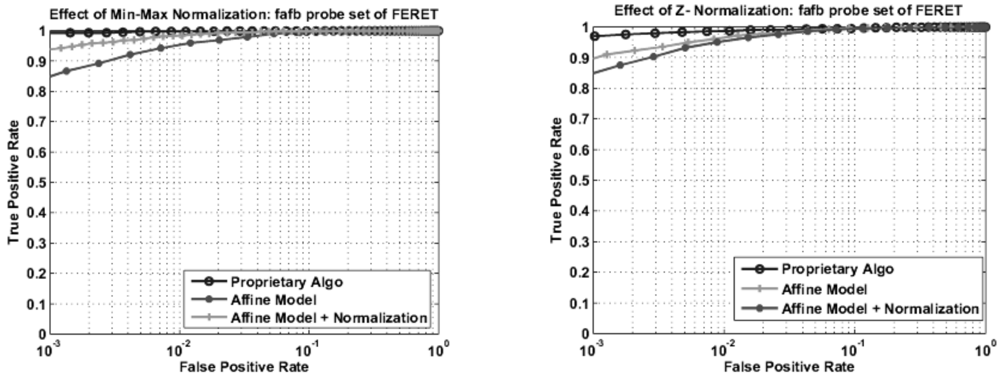
Biometric match scores are often augmented with some normalization procedures before compelled to a threshold based decision. Most of these score normalization techniques are often carried out as a post processing routine and do affect the underlying manifold of the faces as observed by the face recognition algorithms. The most standard score normalization techniques used in biometric applications are *Z-normalization* and *Min-Max normalization* [1], [32], [33]. To observe the impact of normalization on the modeling scheme, we use the proprietary algorithm with min-max and z-normalization techniques. This is over and above any normalization that might exist in the propriety algorithm, about which we do not have any information. We apply the

TABLE III

EFFECT OF DISTANCE MEASURE ON MODEL : ERROR IN MODELING FOR PCA ALGORITHM ON THE FERET FAFB (1195 SUBJECTS) PROBE SET

Distance Measure	**False Acceptance Rate (%)**		
	0.001	**0.01**	**0.1**
CityBlock	0.022	0.018	0.007
Euclidean	0.003	0.0001	0.0001
Correlation	0.004	0.015	0.0005
Covariation	0.003	0.016	0.0002
L1 Norm	0.024	0.003	0.005

normalization methods on impostor scores. Note that, in this case, the normalization techniques are considered as part of the black box algorithm. As a result, the match scores used to train the model are also normalized in the similar way. Fig 13 shows the comparison of recognition performance of the proprietary algorithm with score normalization to that of modeling. The score normalization process is a post processing method and does not reflect the original manifold of the face images. We apply the same score normalization techniques to match scores of the model. The difference between the algorithm with normalized match score and the model with the same normalization of match scores is small.



(a) (b)

Fig. 13. ROC curves indicating score normalization effect on the proposed modeling scheme. We use the proprietary algorithm with two different normalization schemes: (a) Min-Max normalization (b) Z-normalization techniques on the FERET fafb probe set (1195 subjects in probe set) and compare the recognition performance with the performance of the model and performance of the model with similar normalization scheme.

VII. An Application: Indexing Face Databases

In the identification scenario one has to perform one to many matches to identify a new face image (query) among a set of gallery images. In such scenarios, the query image needs to be compared to all the images in gallery. Consequently, the response time for a single query image is directly proportional to the gallery size. The entire process is computationally expensive for large gallery sets. One possible approach to avoid such expensive computation and to provide faster response time is to index or bin the gallery set. In case of well developed biometrics such as fingerprints, a binning process based on ridge patterns such as whorl loop and arches is used for indexing [34], [35]. For other biometrics where a template is represented by a set of d-dimensional numeric features, Mhatre *et al.* [36] proposed a pyramid indexing technique to index the database. Unfortunately, for face images, there is no straightforward and global solution to bin or index face images. As different algorithms use different strategies to compute the template or features from face images, a global indexing strategy is not feasible for face images. For example, Bayesian intra/extra class approach computes the difference image of the probe template with all the gallery templates, a feature based indexing scheme is not applicable for this algorithm.

One possible indexing approach is to use a light or less computationally expensive recognition algorithm to select a subset of gallery images and then compare the probe image with the subset of gallery images. We can project a given probe image into a linear space and find the k nearest gallery images. Then we use the original algorithm to match the k-selected gallery image with probe image and output the rank of the probe image. Note that, for a perfect indexing, a system with indexing and without indexing will produce the same top-k subjects. A linear projection method such as PCA is an example of such first pass pruning method.

The recognition performance of the original algorithm should be better than the first pass linear projection method. Otherwise, the use of a computationally expensive algorithm in the second pass is redundant. Also, if the performance of the first pass algorithm is significantly less than the original algorithm, then the k-gallery image selected by the linear algorithm may not include the nearest gallery images to the probe images as observed by the original algorithm. In this case, the overall identification rate of the system will fall. To minimize this error, the value of k need to be high, which in turn, reduces the advantages of using an indexing mechanism.

On the other hand, since the linear model approximates the underlying algorithm quite well, we expect that basing an indexing scheme around it should result in a better indexing mechanism. The computation complexity for the modeling scheme and any other linear projection based indexing scheme such as PCA is similar, except the training process, which can be done off-line. Of course, for algorithms such as PCA, LDA and ICA which use the linear projection of raw template, this type of indexing mechanism will result in no additional computational advantage. However, for algorithms such as the Bayesian and EBGM, where numerical indexing of template is not feasible, indexing through a linear model can reduce the overall computational complexity by selecting only a subset of gallery images to be matched with a probe image. In this section we have demonstrate the indexing scheme using the proposed linear model and compare an indexing scheme based on PCA, coupled with Euclidean distance. The choice of Euclidean distance instead of Mahalanobis distance is to demonstrate the indexing scenario when the first pass linear projection algorithm has lower performance than the original algorithm.

TABLE IV

Indexing Error Table: Value of *k* at three different indexing error rate for Rank 1 (Rank 5) identification rate on the FERET fafb (1195 Subjects) probe set

(a) Indexing using PCA projection

Algorithm	**Error in Indexing**			
	0.1	**0.01**	**0.001**	**0.0001**
BAY	1(5)	2(10)	13 (18)	13 (75)
EBGM	1 (5)	15(57)	23 (125)	23 (128)
PRP	3 (5)	117 (127)	345 (481)	498 (498)

(b) Indexing using Linear Model

Algorithm	**Error in Indexing**			
	0.1	**0.01**	**0.001**	**0.0001**
BAY	1 (5)	1 (7)	1 (12)	1 (12)
EBGM	1 (5)	9 (15)	43 (43)	48 (48)
PRP	1 (5)	18 (22)	46 (46)	50 (50)

To evaluate the error in the indexing scheme, we use the difference in rank values for a given probe set with and without the indexing scheme. If the model extracts the same *k* nearest gallery image as by the original algorithm, then the rank of a particular probe will not change with the use of the indexing procedure. In such cases, the identification rate at a particular rank will remains the same. However, if the *k*-nearest gallery subjects selected by the model do not match with the *k* nearest subjects selected by the original algorithm, then identification rate at a particular rank will decrease. We compute the error in indexing scheme as follows

$$E_r = \frac{max(I_r - \tilde{I}_r, 0)}{I_r} \tag{18}$$

TABLE V

INDEXING ERROR TABLE: VALUE OF k AT THREE DIFFERENT INDEXING ERROR RATE FOR RANK 1 (RANK 5) IDENTIFICATION RATE ON THE FERET DUP1 (722 SUBJECTS) PROBE SET

(a) Indexing using PCA Projection

Algorithm	**Error in Indexing**			
	0.1	**0.01**	**0.001**	**0.0001**
BAY	1 (5)	5 (19)	16 (27)	16 (35)
EBGM	7 (11)	38 (43)	99 (70)	99 (70)
PRP	5 (21)	18 (75)	22(127)	22 (127)

(b) Indexing using Linear Model

Algorithm	**Error in Indexing**			
	0.1	**0.01**	**0.001**	**0.0001**
BAY	1 (5)	1 (5)	1 (9)	8 (13)
EBGM	1 (5)	15 (15)	20 (22)	20 (22)
PRP	1 (5)	22 (38)	22 (50)	26 (50)

where E_r represents the error in indexing approach at rank r, I_r represents the identification rate of the algorithm at rank r without using indexing of gallery set, and $\tilde{I}_r$ represents the identification rate of the algorithm at rank r using indexing scheme. Note that, if a probe image has a rank higher than k, then we penalize the indexing scheme by setting the rank $\tilde{I}_r$ to 0; ensuring highest possible value of E_r. The maximum is taken to avoid penalizing the indexing scheme in cases where indexing of gallery images yields better identification rate than the original algorithm. (e.g. cases where model of an algorithm has a better recognition rate than the original algorithm). In Table IV and Table V, we show the values of the indexing parameter k at three different indexing error rates for rank 1 and upto rank 5 identification, using both fafb and dup1 probe set, respectively. These two probe sets in the FERET database have maximum number of probe subjects compared to other probe sets. The Table IV(a) and Table V(a) shows the value of k with PCA-based two pass indexing mechanism.

For model based indexing scheme, as we observe, the value of the indexing parameter for the Bayesian algorithm is as low as 8 with error in indexing equals to 0.01%. As a result, with the help of the proposed indexing scheme, the Bayesian algorithm requires at most eight comparisons to achieve similar rank-1 performance as compared to using the complete gallery set, which requires 1195 comparison in the case of the FERET-fafb probe set. Similarly, for the other two algorithms, at most 50 comparisons are sufficient to achieve similar identification performance at 0.01% error rate for rank-1 as well as rank-5 identification performances. With this indexing scheme, the response time is reduced by a factor of $\frac{C_a}{C_m} + \frac{k}{N}$, where C_a and C_m is the time required to match two face images using the original algorithm and its linear model respectively.

N represent the number of gallery images. Since, the proposed modeling scheme requires only a linear projection of face images, in most cases (such as BAY and EBGM algorithms) $C_a \geq C_m$, and $\frac{C_a}{C_m} + \frac{k}{N} << 1$. However, for algorithm such as PCA, LDA and ICA which uses the linear projection of raw template, the model will not provide any computational advantage as in these cases $C_a = C_m$.

In case of PCA-based indexing mechanism, we can observe a high variation in value of indexing parameter k. The indexing performance of PCA-based indexing mechanism on FERET-fafb probe set for the Bayesian and EBGM algorithm is consistent with that of linear modeling index scheme. However, in all other cases, particularly in the case of PRP algorithm, the value of k observed to be very high due to significant performance different between the PCA and the PRP algorithm. Similar values of k is observed even if we use the Mahalanobis distance instead of Euclidean distance for PCA based indexing scheme with PRP algorithm as well. For the PRP algorithm on FERET-fafb probe set, the values of k are 2 (5), 48 (52), 198 (199) and 272 (298) for rank-1 and rank-5 error rates respectively, using indexing with PCA with Mahalanobis distance measure as the first pass pruning method. Similarly, FERET-fafb probe set, the values of k are 2 (8), 20 (44), 26(56) and 28 (57) for rank-1 and rank-5 error rates respectively. These results validate the advantages of using a linear model instead of any arbitrary linear projection method for selecting the k-nearest gallery images in the first pass.

VIII. Conclusion

We proposed a novel, linear modeling scheme for different face recognition algorithms based on the match scores. Starting with a distance matrix representing the pairwise match scores between face images, we used an iterative stress minimization algorithm to obtain an embedding of distance matrix in a low dimensional space. We then proposed a linear out-of-sample projection scheme for test images. The linear transformation used to project new face images into the model space is divided into two sub transformation: a rigid transformation of face images obtained through principal component analysis of face images followed by a non-rigid transformation responsible for preserving pair-wise distance relationships between face images. To validate the proposed modeling scheme, we used six fundamentally different face recognition algorithms, covering both template based and feature based approaches, on four different probe sets using the FERET face image database. We compared the recognition rate of each of the algorithms with their respective models and demonstrated that the recognition rates are consistent on each of the probe set. Experimental results showed that the proposed linear modeling scheme generalized to different probe set representing different variations in face images (FERET probe sets). A 6.3% average error in modeling for six algorithms is observed at 0.001 False Acceptance Rate (FAR), for the FERET fafb probe set which contains maximum number of subjects among all the probe sets. The estimated linear approximation also exhibited an average of 87% match in the nearest neighbor identity with the original algorithms. We also demonstrated the usefulness of such a modeling scheme on algorithm specific indexing of face databases. Although choice of distance measure varied from algorithm to algorithm, we showed that such variations in distance measures have less impact on our proposed modeling scheme. Similarly, many biometric systems uses score normalization as a post processing routine and we observed that similar score normalization routine when applied to match scores obtained through the affine model of the algorithm yields expected recognition performances.

With the help of the proposed modeling scheme, future research will explore the possibility of finding optimal performance of any face recognition algorithm with respect to a given training set. Also, instead of classical scaling other possible choices to arrive at the MDS coordinates include metric least-square scaling that allowed for metric transformations of the given dissimilarities so as to minimize a given loss-function, capturing the differences, maybe weighted, between the transformed dissimilarities and the distances in the embedded space. Note that "metric" in metric scaling refers to the transformation and not the point configuration space. In non-metric scaling, arbitrary, monotonic, transformations are allows as long as rank orders are preserved. These could be focus of future work. However, as we have seen, the stress minimization along with classical MDS suffices to build the linear model for most face recognition algorithms. There is also the danger that complicated schemes might over-fit to the given distances.

REFERENCES

[1] A. K. Jain and S. Li, *Handbook of Face Recognition*. Springer, 2005.

[2] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, “Face recognition: A literature survey,” *ACM Computing Surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[3] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, “The FERET database and evaluation procedure for face recognition algorithms,” in *Image and Vision Computing*, vol. 16, 1998, pp. 295–306.

[4] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, “The FERET evaluation methodology for face-recognition algorithms,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090–1104, 2000.

[5] P. J. Phillips, P. Flynn, T. Scruggs, K. Bowyer, J. Chang, K. Hoffman, J. Marques, J. Min, and W. Worek, “Overview of the face recognition grand challenge,” in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 947–954.

[6] P. J. Phillips, P. Flynn, T. Scruggs, K. Bowyer, and W. Worek, “Preliminary face recognition grand challenge results,” in *International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 15–24.

[7] P. J. Phillips, P. Grother, R. J. Micheals, D. M. Blackburn, E. Tabassi, and M. Bone, “Face recognition vendor test 2002,” in *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, 2003.

[8] P. J. Phillips, W. T. Scruggs, A. J. OToole, P. J. Flynn, K. W. Bowyer, C. L. Schott, and M. Sharpe, “FRVT 2006 and ICE 2006 large-scale results,” in *National Institute of Standards and Technology Internal Report 7408*, 2007.

[9] P. Mohanty, S. Sarkar, and R. Kasturi, “From scores to face template: A model-based approach,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2065–2078, 2007.

[10] T. Cox and M. Cox, *Multidimensional Scaling*, 2nd ed. Chapman and Hall, 1994.

[11] I. Borg and P. Groenen, *Modern Multidimensional Scaling*. Springer Series in Statistics, Springer, 1997.

[12] M. A. Turk and P. Pentland, “Face recognition using eigenfaces,” in *IEEE Conference on Conference on Computer Vision and Pattern Recognition*, 1991, pp. 586–591.

[13] P. Belhumeur, J. Hespanha, and D. Kriegman, “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[14] B. Moghaddam and A. Pentland, “Beyond eigenfaces: Probabilistic matching for face recognition,” in *International Conference on Automatic Face and Gesture Recognition*, 1998, pp. 30–35.

[15] L. Wiskott, J. Fellous, N. Kruger, and C. Malsburg, “Face recognition by elastic bunch graph matching,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 2, no. 7, pp. 1160–1169, 1997.

[16] P. Comon, “Independent component analysis, a new concept?” *Signal Processing*, vol. 36, no. 3, pp. 287–314, 1994.

[17] R. Beveridge, D. Bolme, M. Teixeira, and B. Draper, “The CSU face identification evaluation system,” *Machine Vision and Applications*, vol. 16, no. 2, pp. 128–138, 2005.

[18] E. Pekalska, P. Paclik, and R. P. W. Duin, “A generalized kernel approach to dissimilarity based classification,” *Journal of Machine Learning Research*, vol. 2, pp. 175–211, 2001.

[19] V. Roth, J. Laub, M. Kawanabe, and J. M. Buhmann, “Optimal cluster preserving embedding of nonmetric proximity data,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1540–1551, 2003.

[20] P. Wang, Q. Ji, and J. L. Wayman, “Modeling and predicting face recognition system performance based on analysis of similarity scores,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 665–670, 2007.

[21] S. Mitra, M. Savvides, and A. Brockwell, “Statistical performance evaluation of biometric authentication systems using

random effects models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 517–530, 2007.

[22] R. Wang and B. Bhanu, "Learning models for predicting recognition performance," in *IEEE International Conference on Computer Vision (ICCV)*, 2005, pp. 1613–1618.

[23] G. H. Givens, J. R. Beveridge, B. A. Draper, and P. J. Phillips, "Repeated measures glmm estimation of subject-related and false positive threshold effects on human face verification performance," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*, 2005, p. 40.

[24] P. Grother and P. J. Phillips, "Models of large population recognition performance," in *In Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004, pp. 68–75.

[25] M. Boshra and B. Bhanu, "Predicting performance of object recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 9, pp. 956–969, 2000.

[26] D. J. Litman, J. B. Hirschberg, and M. Swerts, "Predicting automatic speech recognition performance using prosodic cues," in *First conference on North American chapter of the Association for Computational Linguistics*, 2000, pp. 218–225.

[27] J. Gower and P. Legendre, "Metric and Euclidean properties of dissimilarity coefficients," *Journal of Classification*, vol. 3, pp. 5–48, 1986.

[28] E. Pekalska and P. W. Duin, *The Dissimilarity Representation for Pattern Recognition: Foundations and Applications*, 1st ed., ser. Series in Machine Perception and Artificial Intelligence. World Scientific, 2006, vol. 64.

[29] M. Bartlett, *Face Image Analysis by Unsupervised Learning*. Kluwer Academic Publishers, 2001.

[30] M. L. Teixeira, "The Bayesian intrapersonal/extrapersonal classfier," in *Masters Thesis, Colorado State University*, 2003.

[31] D. Bolme, "Elastic bunch graph matching," in *Masters Thesis, Colorado State University*, 2003.

[32] S. Prabhakar and A. K. Jain, "Decision-level fusion in fingerprint verification," *Pattern Recognition*, vol. 35, no. 4, pp. 861–874, 2002.

[33] J. Kittler, M. Hatef, R. P. Duin, and J. G. Matas, "Decision-level fusion in fingerprint verification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[34] R. Cappelli, D. Maio, D. Maltoni, and L. Nanni, "A two-stage fingerprint classification system," in *ACM SIGMM workshop on Biometrics methods and applications*, 2003, pp. 95–99.

[35] N. Ratha, K. Karu, S. Chen, and A. Jain, "A real-time matching system for large fingerprint databases," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 799–813, 1996.

[36] A. Mhatre, S. Palla, S. Chikkerur, and V. Govindaraju, "Efficient search and retrieval in biometric databases"," *SPIE Defense and Security Symposium*, vol. 5779, pp. 265–273, 2005.