Linear Discriminant Analysis for Subclustered Data

Jaegul Choo*, Barry L. Drake†, and Haesun Park*

*College of Computing, Georgia Institute of Technology, 266 Ferst Drive, Atlanta, GA 30332, USA, {joyfull, hpark}@cc.gatech.edu

†SEAL/AMDD, Georgia Tech Research Institute, 7220 Richardson Road, Smyrna, GA 30080, USA, barry.drake@gtri.gatech.edu

The work of these authors was supported in part by the National Science Foundation grants CCF-0621889 and CCF-0732318. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Abstract

Linear discriminant analysis (LDA) is a widely-used feature extraction method in classification. However, the original LDA has limitations due to the assumption of a unimodal structure for each cluster, which is not satisfied in many applications such as facial image data when variations, e.g. angle and illumination, can significantly influence the images. In this paper, we propose a novel method called hierarchical LDA (h-LDA), which takes into account hierarchical subcluster structures of the data in the LDA formulation and algorithm. We develop a theoretical basis of hierarchical LDA by identifying its relation to two-way multivariate analysis of variance (MANOVA) based on the data model and variance decomposition. Furthermore, an efficient algorithm for a regularized version of h-LDA (h-RLDA) is presented using the QR decomposition and the generalized SVD. To validate the effectiveness of the proposed method, we compare face recognition performance among h-RLDA, LDA, PCA, and TensorFaces. Our experiments show that h-RLDA produces better prediction accuracy than other methods. When only a small subset of features are used (reduced dimensionality), the superiority of h-RLDA over other methods becomes more significant. It is also shown that h-RLDA is a computationally much more efficient alternative to TensorFaces.

Index Terms

Dimension reduction, Feature extraction, Generalized singular value decomposition, QR decomposition, Hierarchical clustering, Undersampled problem, Regularization, Face recognition, Classification

I. INTRODUCTION

Linear discriminant analysis (LDA) has been one of the most widely-used dimension reduction methods for classification problems over the past several decades. LDA provides an optimal linear transformation into a lower dimensional space that preserves the cluster information. In facial recognition applications, LDA has also proven its capability even when raw pixel values are used as a feature vector without applying sophisticated feature encoding schemes [1].

Classical LDA relies on the nonsingularity of the scatter matrices, where the number of data must be greater than the dimensionality, which we call an oversampled case. However, many modern data sets such as image data are undersampled, and in order to mitigate the undersampled problem, preprocessing steps such as principal component analysis (PCA) can be applied prior to LDA [2]–[4]. On the other hand, Park et al. devised an algorithm to directly

solve the undersampled LDA problem without any additional preprocessing steps by applying the generalized singular value decomposition (GSVD) [1, 5], and also proposed its efficient and robust version by applying QR decomposition and regularization [6].

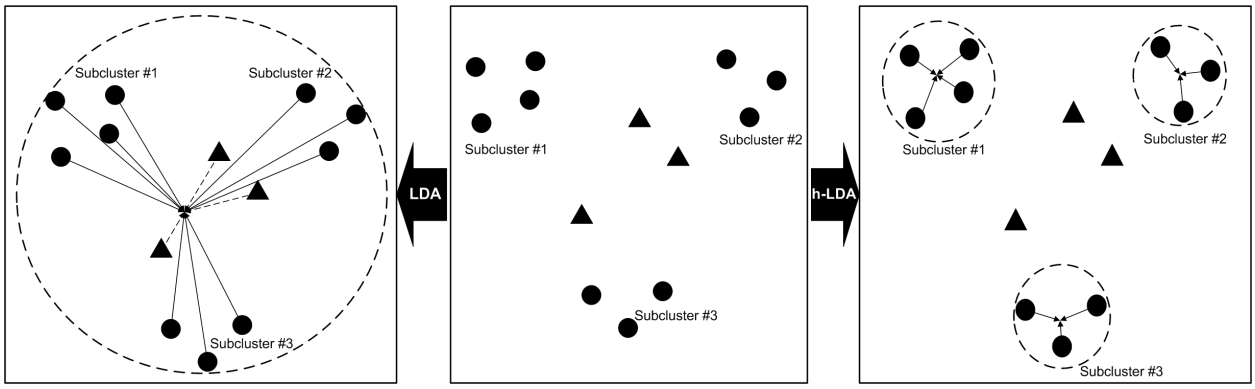
One of the important limitations of LDA is the assumption that each class is modeled as a unimodal Gaussian that can be fully described only with the first and the second order statistics, i.e. mean and covariance. In reality, there is no guarantee that the data conforms to such assumptions. Furthermore, when the data is significantly dependent on other factors than the cluster label of interest, the data corresponding to a particular label may not be simply modeled as a unimodal Gaussian with a single mean and a covariance. Also, in many applications such as face recognition, often the factors such as pose and/or illumination produces noticeably different facial images of the same person.

In order to circumvent such problems, one may apply several variants such as nonparametric discriminant analysis (NDA) [7] , subclass discriminant analysis (SDA) [8], or regularized LDA [6, 9]. NDA uses a nonparametric form of the between-cluster scatter matrix to relax the unimodal Gaussian assumption. SDA applies the multi-modal Gaussian model directly by replacing each cluster centroid with subcluster centroids in the definition of between-cluster scatter matrix. Although regularized LDA was originally introduced for the purpose of avoiding singularity of the within-cluster scatter matrix for undersampled cases, regularization also controls overfitting by preventing the within-cluster relationship from becoming too tight. Such regularization can also be viewed as suppressing the effect of the off-diagonal components in the within-cluster scatter matrix, or equivalently, as imposing the identity covariance matrix as a prior form in a Bayesian sense.

However, other information can be utilized in addition to the cluster label. In face recognition, there have been several approaches where various information such as angle, illumination, and/or pose of images are utilized so that recognition performance can be enhanced. TensorFaces [10] has shown its advantages over PCA in face recognition by constructing tensor data structures based on all the available factors and then by applying N-mode SVD [10].

In this paper, we propose a novel method called hierarchical LDA (h-LDA) that formulates a new feature extraction method based on hierarchical cluster structure of depth two in the data. In h-LDA, clusters are composed of several subclusters determined by other factors than the cluster label of interest. The data points in a cluster do not have to be close, as in the unimodal Gaussian

Fig. 1. Assume that the data points belong to two clusters: the triangular points belong to one cluster and the circular points to the other cluster which can be further clustered into three subclusters. LDA may represent the data from these two clusters close to each other as a result of minimizing the within-cluster relationship among circular points while h-LDA can avoid this problem by *emphasizing within-subcluster structure*.



model, if subclusters are well-separated in the original feature space. In this case, if the data from other clusters happen to be located in the area surrounded by those subclusters, the classical LDA would put them close to each other in the reduced dimensional space, deteriorating the cluster separability (see Figure 1). The criterion of h-LDA decomposes the within-cluster scatter matrix at a subcluster level, and enables us to adjust the relative weights of the decomposed subcluster scatter matrices. Thus, avoiding subcluster mixing.

The rest of this paper is organized as follows. In Section 2, the classical LDA is briefly reviewed, and h-LDA for data with hierarchical cluster structure is introduced in Section 3. In Section 4, its relationship to two-way multivariate analysis of variance (MANOVA) in the context of hypothesis testing is shown, and in Section 5, a regularized version of h-LDA is proposed. Experimental results on the Shimon Edelman's face database are reported in Section 6, and finally conclusions are given in Section 7.

II. Linear Discriminant Analysis

In LDA, an optimal dimension-reduced representation of data is obtained by a linear transformation that maximizes the *conceptual* ratio of the between-cluster scatter (variance) versus the within-cluster scatter of the data. In this section, we present an overview of the basic ideas of LDA. For more details, refer to [11, 12].

Given a data matrix $A = [a_1\, a_2 \cdots a_n] \in \mathbb{R}^{m\times n}$, where n columns a_i, $i = 1, \ldots, n$, of A represent n data items in an m dimensional space, assume that the columns of A are partitioned

into p clusters as

$$A = [A_1 \quad A_2 \quad \cdots \quad A_p],$$

where

$$A_i \in \mathbb{R}^{m \times n_i} \text{ and } \sum_{i=1}^{p} n_i = n.$$

Let $\mathcal{N}_i$ denote the set of column indices that belong to cluster i, n_i the size of $\mathcal{N}_i$, a_k the data point represented in the k-th column vector of A, $c^{(i)}$ the centroid of the i-th cluster, and c the global centroid. For example, in face recognition, A_i corresponds to the set of images of the i-th person.

The scatter matrix within the i-th cluster $S_w^{(i)}$, the within-cluster scatter matrix S_w, the between-cluster scatter matrix S_b, and the total (or mixture) scatter matrix S_t, are defined as

$$S_w^{(i)} = \sum_{k \in \mathcal{N}_i} (a_k - c^{(i)})(a_k - c^{(i)})^T, \tag{1}$$

$$S_w = \sum_{i=1}^{p} S_w^{(i)} = \sum_{i=1}^{p} \sum_{k \in \mathcal{N}_i} (a_k - c^{(i)})(a_k - c^{(i)})^T, \tag{2}$$

$$\begin{aligned} S_b &= \sum_{i=1}^{p} \sum_{k \in \mathcal{N}_i} (c^{(i)} - c)(c^{(i)} - c)^T \\ &= \sum_{i=1}^{p} n_i (c^{(i)} - c)(c^{(i)} - c)^T, \text{ and} \end{aligned} \tag{3}$$

$$\begin{aligned} S_t &= \sum_{k=1}^{n} (a_k - c)(a_k - c)^T \\ &= S_w + S_b, \end{aligned} \tag{4}$$

respectively [2, 13].

In addition, we can also define virtual "square-root" factors H_w, H_b, and H_t of S_w, S_b, and S_t, respectively, as

$$H_w = [A_1 - c^{(1)}e^{(1)^T}, A_2 - c^{(2)}e^{(2)^T}, \ldots, A_p - c^{(p)}e^{(p)^T}] \in \mathbb{R}^{m \times n}, \tag{5}$$

$$H_b = [\sqrt{n_1}(c^{(1)} - c), \sqrt{n_2}(c^{(2)} - c), \ldots, \sqrt{n_p}(c^{(p)} - c)] \in \mathbb{R}^{m \times p}, \text{ and} \tag{6}$$

$$H_t = [a_1 - c, \ldots, a_n - c] = A - ce^T \in \mathbb{R}^{m \times n}, \tag{7}$$

where $e^{(i)} \in \mathbf{R}^{n_i \times 1}$ and $e \in \mathbf{R}^{n \times 1}$ are column vectors where all components are 1's. Then the scatter matrices can be expressed as

$$S_w = H_w H_w^T,\ S_b = H_b H_b^T, \text{ and } S_t = H_t H_t^T, \tag{8}$$

and

$$trace(S_w) = \sum_{i=1}^{p} \sum_{j \in \mathcal{N}_i} \|a_j - c^{(i)}\|_2^2 \text{ and } trace(S_b) = \sum_{i=1}^{p} \sum_{j \in \mathcal{N}_i} \|c^{(i)} - c\|_2^2. \tag{9}$$

In the lower dimensional space obtained by a linear transformation

$$G^T : x \in \mathbb{R}^{m \times 1} \rightarrow y \in \mathbb{R}^{l \times 1}, \tag{10}$$

the within-cluster, the between-cluster, and the total scatter matrices become

$$S_w^Y = G^T S_w G, \quad S_b^Y = G^T S_b G, \quad \text{and} \quad S_t^Y = G^T S_t G,$$

where the superscript Y denotes the scatter matrices in the l dimensional space obtained by applying G^T. In LDA, an optimal linear transformation matrix G^T is found so that it minimizes the within-cluster scatter measure $trace(S_w^Y)$ and at the same time, maximizes the between-cluster scatter measure $trace(S_b^Y)$. This optimization problem of two distinct measures is usually replaced with one that maximizes

$$J(G) = trace((G^T S_w G)^{-1}(G^T S_b G)). \tag{11}$$

Assuming $S_w = H_w H_w^T$ is nonsingular, it can be shown that [5, 14]

$$trace((S_w^Y)^{-1} S_b^Y) \leq trace(S_w^{-1} S_b) = \sum_i \lambda_i,$$

where λ_i's are the eigenvalues of $S_w^{-1} S_b$. The upper bound on $J(G)$ is achieved as

$$\max_G trace((S_w^Y)^{-1} S_b^Y) = trace(S_w^{-1} S_b)$$

when $G \in \mathbb{R}^{m \times l}$ consists of l eigenvectors of $S_w^{-1} S_b$ corresponding to the l largest eigenvalues in the eigenvalue problem

$$S_w^{-1} S_b x = \lambda x, \tag{12}$$

where l is the number of nonzero eigenvalues of $S_w^{-1} S_b$. Since the rank of S_b is at most $p-1$, if we set $l = p - 1$, and solve for G from Eq. (12), then we can obtain the best dimension reduction that does not lose the cluster separability measured by $trace(S_w^{-1} S_b)$.

Algorithm 1 LDA/GSVD

Given a data matrix $A \in \mathbb{R}^{m\times n}$ where the columns are partitioned into p clusters, this algorithm computes the dimension reducing transformation $G \in \mathbb{R}^{m\times(p-1)}$. For any vector $x \in \mathbb{R}^{m\times 1}$, $y = G^T x \in \mathbb{R}^{(p-1)\times 1}$ gives a $(p-1)$ dimensional representation of x.

1) Compute $H_b \in \mathbb{R}^{m\times p}$ and $H_w \in \mathbb{R}^{m\times n}$ from A according to Eq. (6) and (5), respectively.
2) Compute the complete orthogonal decomposition of $K = \begin{pmatrix} H_b^T \\ H_w^T \end{pmatrix} \in \mathbb{R}^{(p+n)\times m}$, i.e., $P^T K V = \begin{pmatrix} R & 0 \\ 0 & 0 \end{pmatrix}$, where $P \in \mathbb{R}^{(p+n)\times(p+n)}$ and $V \in \mathbb{R}^{m\times m}$ are orthogonal matrices, and R is a square matrix with $rank(K) = rank(R)$.
3) Let $t = rank(K)$.
4) Compute W from the SVD of $P(1:p, 1:t)$, i.e., $U^T P(1:p, 1:t) W = \Sigma$.
5) Compute the first $p-1$ columns of $V \begin{pmatrix} R^{-1}W & 0 \\ 0 & I \end{pmatrix}$, and assign them to G.

One limitation of using the criteria $J(G)$ is that S_w must be invertible. However, in many applications including face recognition, the dimensionality m is often much greater than the number of data n, making S_w singular. Expressing λ as α^2/β^2, and using Eq. (8), Eq. (12) can be rewritten as

$$\beta^2 H_b H_b^T x = \alpha^2 H_w H_w^T x. \tag{13}$$

Then, this reformulation turns out to be a generalized singular value decomposition (GSVD) problem [15]–[17], and it can give the solution of LDA regardless of the singularity of S_w. This GSVD-based LDA algorithm is summarized in Algorithm 1 LDA/GSVD. For more details, see [5, 13].

III. Hierarchical LDA (h-LDA)

An assumption of the classical LDA is that the data distribution in each cluster is a unimodal Gaussian centered at a single centroid. Under this assumption, minimizing the criterion expressed in Eq. (11) gives the optimal solution based on the second order statistics over the data. In many applications, however, the structure of the data cannot be simply explained by this assumption.

Relaxing such a simplified assumption, hierarchical LDA (h-LDA) assumes that the data in cluster i, A_i, can be further clustered into q_i subclusters as

$$A_i = [A_{i1} \quad A_{i2} \quad \cdots \quad A_{iq_i}],$$

where

$$A_{ij} \in \mathbb{R}^{m \times n_{ij}}, \quad \sum_{j=1}^{q_i} n_{ij} = n_i.$$

Let $\mathcal{N}_{ij}$ denote the set of column indices that belong to the subcluster j in cluster i, n_{ij} the size of $\mathcal{N}_{ij}$ and $c^{(ij)}$ the centroid of each subcluster. In facial image data, the set of images of a specific person can be further clustered according to angles of view, or illumination conditions for example. Then, we can define the scatter matrix within subcluster j of cluster i, $S_{w_s}^{(ij)}$, their sum in cluster i, $S_{w_s}^{(i)}$, and the scatter matrix between subclusters in cluster i, $S_{b_s}^{(i)}$, respectively, as

$$S_{w_s}^{(ij)} = \sum_{k \in \mathcal{N}_{ij}} (a_k - c^{(ij)})(a_k - c^{(ij)})^T, \tag{14}$$

$$S_{w_s}^{(i)} = \sum_{j=1}^{q_i} S_{w_s}^{(ij)} = \sum_{j=1}^{q_i} \sum_{k \in \mathcal{N}_{ij}} (a_k - c^{(ij)})(a_k - c^{(ij)})^T, \text{ and} \tag{15}$$

$$\begin{aligned} S_{b_s}^{(i)} &= \sum_{j=1}^{q_i} \sum_{k \in \mathcal{N}_{ij}} (c^{(ij)} - c^{(i)})(c^{(ij)} - c^{(i)})^T \\ &= \sum_{j=1}^{q_i} n_{ij} (c^{(ij)} - c^{(i)})(c^{(ij)} - c^{(i)})^T. \end{aligned} \tag{16}$$

Then, the within-subcluster scatter matrix S_{w_s} and the between-subcluster scatter matrix S_{b_s} are defined respectively as

$$\begin{aligned} S_{w_s} &= \sum_{i=1}^{p} S_{w_s}^{(i)} = \sum_{i=1}^{p} \sum_{j=1}^{q_i} S_{w_s}^{(ij)} \\ &= \sum_{i=1}^{p} \sum_{j=1}^{q_i} \sum_{k \in \mathcal{N}_{ij}} (a_k - c^{(ij)})(a_k - c^{(ij)})^T, \end{aligned} \tag{17}$$

$$\begin{aligned} S_{b_s} &= \sum_{i=1}^{p} S_{b_s}^{(i)} \\ &= \sum_{i=1}^{p}\sum_{j=1}^{q_i}\sum_{k\in\mathcal{N}_{ij}} (c^{(ij)}-c^{(i)})(c^{(ij)}-c^{(i)})^T \\ &= \sum_{i=1}^{p}\sum_{j=1}^{q_i} n_i (c^{(ij)}-c^{(i)})(c^{(ij)}-c^{(i)})^T. \end{aligned} \tag{18}$$

From the identity

$$a_k - c = (a_k - c^{(ij)}) + (c^{(ij)} - c^{(i)}) + (c^{(i)} - c),$$

it can be proved that

$$S_t = S_{w_s} + S_{b_s} + S_b \tag{19}$$

where the between-cluster scatter matrix S_b is defined as in Eq. (3). Comparing Eq. (19) with Eq. (4), the within-cluster scatter matrix S_w in LDA is equivalent to the sum of the within-subcluster scatter matrix S_{w_s} and the between-subcluster scatter matrix S_{b_s} as

$$S_w = S_{w_s} + S_{b_s}. \tag{20}$$

Now we propose a new within-cluster scatter matrix S_w^h, which is a convex combination of S_{w_s} and S_{b_s} as

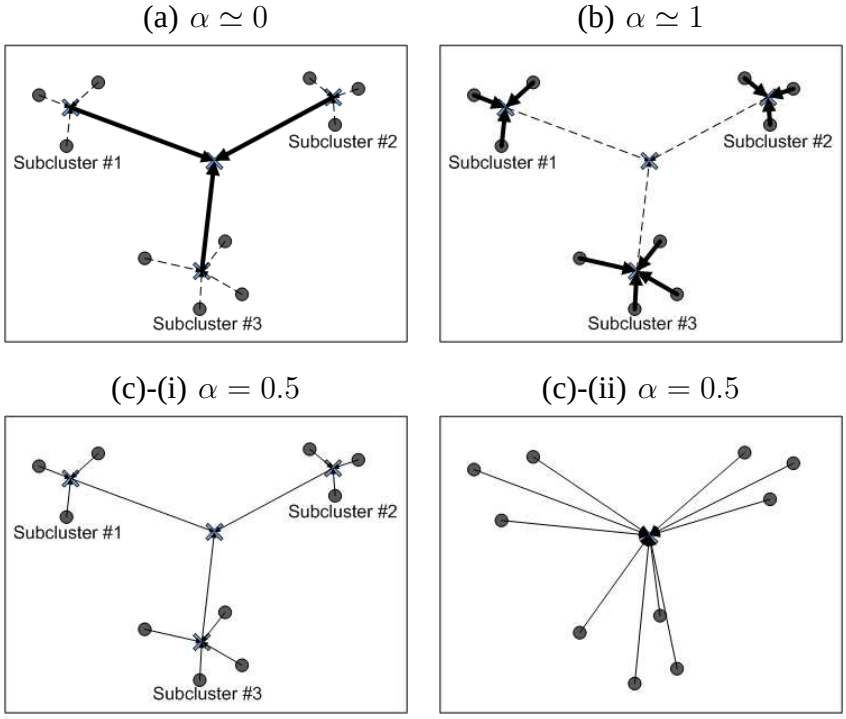
$$S_w^h = \alpha S_{w_s} + (1-\alpha) S_{b_s}, \quad 0 \le \alpha \le 1, \tag{21}$$

where α determines relative weights between S_{w_s} and S_{b_s}. By replacing S_w with the newly-defined S_w^h, h-LDA finds the solution that maximizes the new criterion

$$J^h(G) = trace((G^T S_w^h G)^{-1}(G^T S_b G)). \tag{22}$$

Consider the following three cases: $\alpha \simeq 0$, $\alpha \simeq 1$, and $\alpha = 0.5$. When $\alpha \simeq 0$ (see Figure 2(a)), the within-subcluster scatter matrix S_{w_s} is disregarded and the between-subcluster scatter matrix S_{b_s} is emphasized, which can be considered as the original LDA applied after every data point is relocated to its corresponding subcluster centroid. When $\alpha \simeq 1$ (see Figure 2(b)), h-LDA minimizes only the within-subcluster radii, disregarding the distances between subclusters within each cluster. When $\alpha = 0.5$, the within-subcluster scatter matrix S_{w_s} and the between-subcluster scatter matrix S_{b_s} are equally weighted so that h-LDA becomes equivalent to LDA by Eq. (20), which shows the equivalence of the within-cluster scatter matrices between Figure 2(c)-(i) and 2(c)-(ii). Hence, h-LDA can be viewed as a generalization of LDA, and the parameter α can be

Fig. 2. Example of h-LDA and the parameter α. All data points in each figure belong to one cluster.



chosen by parameter optimization schemes such as cross-validation in order to attain maximum classification performance. Considering the motivation of h-LDA, attention should be payed to the case of $0.5 < \alpha \simeq 1$ since this can mitigate the unimodal Gaussian assumption weakness of the classical LDA, which can produce a transformation that projects the points in one cluster onto essentially one point in the reduced dimensional space.

Based on the LDA/GSVD framework [18], the "square-root" factors H_{w_s}, H_{b_s}, and H_w^h of S_{w_s}, S_{b_s}, and S_w^h, respectively, can also be defined as

$$\begin{aligned} H_{w_s} &= [A_{11} - c^{(11)}e^{(11)^T}, \ldots, A_{1q_1} - c^{(1q_1)}e^{(1q_1)^T}, \\ &\quad A_{21} - c^{(21)}e^{(21)^T}, \ldots, A_{2q_2} - c^{(2q_2)}e^{(2q_2)^T}, \\ &\quad \ldots, A_{p1} - c^{(p1)}e^{(p1)^T}, \ldots, A_{pq_p} - c^{(pq_p)}e^{(pq_p)^T}] \in \mathbb{R}^{m \times n}, \end{aligned} \tag{23}$$

$$\begin{aligned} H_{b_s} &= [\sqrt{n_{11}}(c^{(11)} - c^{(1)}), \ldots, \sqrt{n_{1q_1}}(c^{(1q_1)} - c^{(1)}), \\ &\quad \sqrt{n_{21}}(c^{(21)} - c^{(2)}), \ldots, \sqrt{n_{2q_2}}(c^{(2q_2)} - c^{(2)}), \\ &\quad \ldots, \sqrt{n_{pq_p}}(c^{(pq_p)} - c^{(p)}), \ldots, \sqrt{n_{pq_p}}(c^{(pq_p)} - c^{(p)})] \in \mathbb{R}^{m \times s}, \end{aligned} \tag{24}$$

$$H_w^h = [\sqrt{\alpha}H_{w_s} \ \sqrt{1-\alpha}H_{b_s}] \tag{25}$$

Algorithm 2 h-LDA/GSVD

Given a data matrix $A \in \mathbb{R}^{m\times n}$ where the columns are partitioned into p clusters, and each of them is further clustered into q_i clusters for $i = 1, \ldots, p$, respectively, this algorithm computes the dimension reducing transformation $G \in \mathbb{R}^{m\times(p-1)}$. For any vector $x \in \mathbb{R}^{m\times 1}$, $y = G^T x \in \mathbb{R}^{(p-1)\times 1}$ gives a $(p-1)$ dimensional representation of x.

1) Compute $H_b \in \mathbb{R}^{m\times p}$, $H_{w_s} \in \mathbb{R}^{m\times n}$, and $H_{b_s} \in \mathbb{R}^{m\times s}$ from A according to Eqs. (6), (23), and (24), respectively, where $s = \sum_{i=1}^{p} q_i$.
2) Compute the complete orthogonal decomposition of $K^h = \begin{pmatrix} H_b^T \\ \sqrt{\alpha}H_{w_s}^T \\ \sqrt{1-\alpha}H_{b_s}^T \end{pmatrix} \in \mathbb{R}^{(p+n+s)\times m}$, i.e. $P^T K^h V = \begin{pmatrix} R & 0 \\ 0 & 0 \end{pmatrix}$, where $P \in \mathbb{R}^{(p+n+s)\times(p+n+s)}$ and $V \in \mathbb{R}^{m\times m}$ are orthogonal, and R is a square matrix with $rank(K^h) = rank(R)$.
3) Let $t = rank(K^h)$.
4) Compute W from the SVD of $P(1:p, 1:t)$, i.e., $U^T P(1:p, 1:t)W = \Sigma$.
5) Compute the first $k-1$ columns of $V \begin{pmatrix} R^{-1}W & 0 \\ 0 & I \end{pmatrix}$, and assign them to G.

where $s = \sum_{i=1}^{p} q_i$ and $e^{(ij)} \in \mathbf{R}^{n_{ij}\times 1}$ is a vector where all components are 1's. Then the scatter matrices can be expressed as

$$S_{w_s} = H_{w_s} H_{w_s}^T, \quad S_{b_s} = H_{b_s} H_{b_s}^T, \text{ and } S_w^h = H_w^h (H_w^h)^T. \tag{26}$$

Based on Algorithm 1 for the LDA/GSVD, the h-LDA/GSVD algorithm is designed and summarized in Algorithm 2 h-LDA/GSVD.

IV. Relationship between h-LDA and Two-way MANOVA

Multivariate Analysis of Variance (MANOVA) [19, 20] is a hypothesis testing method that determines whether the data of each cluster is significantly different from each other based on the data distribution, or equivalently, whether the treatment factor that assigns different treatments (or cluster labels) actually indicates a significantly different data distribution depending on the

cluster label. For instance, suppose we have two groups of infant trees and provide only one group with plenty of water. If we observe the heights of the trees a few months later, probably the average heights of the two groups would be noticeably different compared to the variation of heights within each group, and we could conclude that the treatment factor of giving more water has a significant influence on the data. The observed data in this example is just a one dimensional value, i.e. heights, but if the dimensionality of the data becomes larger and the number of clusters increases, then this test would require a more sophisticated measure. This is the main motivation of MANOVA.

To begin with, MANOVA assumes each cluster is modeled as a Gaussian with its own mean vector but with a common covariance matrix. It can be easily seen that the estimates of the within-cluster and the between-cluster covariances correspond to Eq. (2) and Eq. (3) respectively, and Eq. (4) holds accordingly. Among many of MANOVA tests for the significant difference between cluster-wise data distributions, Hotelling-Lawley trace test [21] uses $trace(S_w^{-1}S_b)$ as a cluster separability measure. As shown in Section 2, LDA gives the dimension reduced representation that preserves this measure as in the original space. Therefore, it is interesting to see that although the objective of LDA is different from that of MANOVA based on Hotelling-Lawley trace measure, they are based on the same measure of class separability. Accordingly, the dimension reduction by LDA would not affect MANOVA tests since LDA preserves $trace(S_w^{-1}S_b)$ in the lower dimensional space.

Now we apply a similar analogy to the relationship between h-LDA and two-way MANOVA. Starting from the data model of two-way MANOVA, we derive its variance decomposition, and show the equivalence between the Hotelling-Lawley trace test and the h-LDA criterion. In two-way MANOVA, each datum is assigned a pair of cluster labels, which are determined by two treatment factors. To be more specific in trees case, two treatment factors such as water and light might be considered as potential treatment factors. Depending on whether sufficient water and/or light are provided, the heights of trees are observed as a dependent variable. Two-way MANOVA test determines if each factor has a significant effect on the height of trees as well as if the two factors are independent or not. For instance in face recognition, if the first factor corresponds to person id and the second to angles of view, each image would be given a person id and an angle value as their label pair.

In two-way MANOVA, the k-th data point with its label pair $(i,\ j)$, which corresponds to the

i-th treatment from the first factor and the j-th treatment from the second factor, is modeled as

$$x_k = c + c^{(i\cdot)} + c^{(\cdot j)} + \epsilon_{ij} + \epsilon_k, \tag{27}$$

where c is the global mean, $c^{(i\cdot)}$ for $i = 1, \cdots, p$ is the mean of the data with the i-th treatment from the first factor, $c^{(\cdot j)}$ for $j = 1, \cdots, q$ is the mean of the data with the j-th treatment from the second factor, and ϵ_{ij} and ϵ_k are independent and identically distributed (i.i.d.) zero mean Gaussian random variables. Without loss of generality, we can impose the assumption that $\sum_{i=1}^{p} c^{(i\cdot)} = 0$ and $\sum_{j=1}^{q} c^{(\cdot j)} = 0$. The model in Eq. (27) implies that the cluster mean with label pair (i, j) is represented as an additive model of two independent values, $c^{(i\cdot)}$ and $c^{(\cdot j)}$, with the cluster-wise error term ϵ_{ij}. Then the instance-wise error term ϵ_k is introduced to each datum x_k.

The total scatter matrix S_t, the residual scatter matrix S_r, the interaction scatter matrix S_i, the first factor between-cluster scatter matrix S_{b1} and the second factor between-cluster scatter matrix S_{b2} are defined respectively as

$$\begin{aligned} S_t &= \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k\in\mathcal{N}_{ij}} (a_k - c)(a_k - c)^T \\ &= \sum_{k=1}^{n} (a_k - c)(a_k - c)^T, \end{aligned} \tag{28}$$

$$S_r = \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k\in\mathcal{N}_{ij}} (a_k - c^{(ij)})(a_k - c^{(ij)})^T, \tag{29}$$

$$\begin{aligned} S_a &= \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k\in\mathcal{N}_{ij}} (c^{(ij)} - c^{(i\cdot)} - c^{(\cdot j)} + c)(c^{(ij)} - c^{(i\cdot)} - c^{(\cdot j)} + c)^T \\ &= \sum_{i=1}^{p}\sum_{j=1}^{q} n_{ij}(c^{(ij)} - c^{(i\cdot)} - c^{(\cdot j)} + c)(c^{(ij)} - c^{(i\cdot)} - c^{(\cdot j)} + c)^T, \end{aligned} \tag{30}$$

$$\begin{aligned} S_{b1} &= \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k\in\mathcal{N}_{ij}} (c^{(i\cdot)} - c)(c^{(i\cdot)} - c)^T \\ &= \sum_{i=1}^{p} n_{i\cdot}(c^{(i\cdot)} - c)(c^{(i\cdot)} - c)^T, \end{aligned} \tag{31}$$

$$S_{b2} = \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k\in\mathcal{N}_{ij}}(c^{(.j)}-c)(c^{(.j)}-c)^T$$
$$= \sum_{j=1}^{q} n_{.j}(c^{(.j)}-c)(c^{(.j)}-c)^T. \quad (32)$$

From the above definitions, the total scatter matrix S_t in two-way MANOVA is decomposed as

$$S_t = S_r + S_a + S_{b2} + S_{b1}. \quad (33)$$

Assuming $q_1 = q_2 = \cdots = q_p = q$ in h-LDA, the within-subcluster scatter matrix S_{w_s} in Eq. (17) becomes the same as the residual scatter matrix S_r in Eq. (29). If we view the first factor label i as the cluster label of interest in h-LDA, and equate Eq. (19) and Eq. (33), we obtain

$$S_{b_s} = S_a + S_{b2} \text{ and} \quad (34)$$
$$S_b = S_{b1}. \quad (35)$$

Now in two-way MANOVA, the Hotelling-Lawley trace measures [21] the class separability due to the first and second factors respectively as

$$H_1 = trace(S_{w_s}^{-1}S_{b1}) \text{ and} \quad (36)$$
$$H_2 = trace(S_{w_s}^{-1}S_{b2}). \quad (37)$$

By comparing these measures with the statistically-predetermined thresholds, it is determined whether an observed response to the treatment is statistically significant. Similarly, the Hotelling-Lawley measure determines whether an interaction between two factors exists, i.e. whether two factors are independent of each other, based on

$$H_a = trace(S_{w_s}^{-1}S_a).$$

Comparing Eq. (36) with the h-LDA criterion of Eq. (22), the solution of h-LDA with $\alpha = 1$ gives the optimal linear transformation that preserves the Hotelling-Lawley trace measure H_1 in the two-way MANOVA model. Thus in this particular case of $\alpha = 1$, we can conclude that the underlying data model of h-LDA maintains the additive nature of two independent factors as in Eq. (27).

V. h-LDA WITH REGULARIZATION (h-RLDA)

Both h-LDA and LDA take into account only the estimates of the first and the second order statistics of the data, and the quality of these estimates relies on the number of data items. In other words, as the number of data items increases, the estimators have smaller variances or smaller deviations from the true underlying values. In this sense, the potential drawback in h-LDA is that the estimates of S_{w_s} and S_{b_s} may not be as confident as that of S_w due to further splitting of the data into subclusters. In order to overcome this problem, we propose introducing a regularization term into S_w^h in Eq. (22) as

$$\begin{aligned} J^h(G,\gamma) &= trace((G^T(S_w^h+\gamma I)G)^{-1}(G^T S_b G)) \\ &= trace((G^T(\alpha S_{w_s}+(1-\alpha)S_{b_s}+\gamma I)G)^{-1}(G^T S_b G)), \end{aligned} \tag{38}$$

which enables us to avoid the difficulty resulting from the size of each subcluster being too small by adjusting the value of $\gamma > 0$. The criterion to maximize Eq. (38) is a regularized form of h-LDA, which we call h-RLDA.

Now we propose an algorithm for h-RLDA based on the efficient regularized LDA algorithm that was recently proposed in [6].

Although regularized LDA has been commonly used for dimension reduction of high dimensional data in many applications, high dimensionality can make the time complexity and memory requirements very expensive. To cope with this problem, the algorithms proposed in [6] utilize QR decomposition for undersampled problem or Cholesky decomposition for oversampled problem as a preprocessing step, and these preprocessing steps can reduce the problem size and accordingly the computational complexity dramatically. We discuss our algorithm in detail for undersampled case, i.e., when the dimensionality is higher than the number of the data. The oversampled case is analogous.

For any matrix $A \in \mathbb{R}^{m\times n}$ with $m \geq n$, there exists an orthogonal matrix $Q \in \mathbb{R}^{m\times m}$ and an upper triangular matrix $R \in \mathbb{R}^{n\times n}$ such that

$$A = Q\begin{pmatrix} R \\ 0_{(m-n)\times n} \end{pmatrix} = \begin{pmatrix} Q_1 & Q_2 \end{pmatrix}\begin{pmatrix} R \\ 0_{(m-n)\times n} \end{pmatrix} = Q_1 R$$

where $Q_1 \in \mathbb{R}^{m\times n}$ and $Q_2 \in \mathbb{R}^{m\times(m-n)}$. Then we have

$$Q_1^T A = R, \tag{39}$$

and R can be partitioned into k clusters $[R_1 \cdots R_p]$ as in A, and R_i can be further partitioned into their subclusters $[R_{i1} \cdots R_{iq_i}]$ for $i = 1, \ldots, p$ where $R_i = Q_1^T A_i \in \mathbb{R}^{n \times n_i}$ and $R_{ij} = Q_1^T A_{ij} \in \mathbb{R}^{n \times n_{ij}}$ for $i = 1, ..., p$ and $j = 1, \ldots, q_i$. Now the following matrices are formed using R as

$$\begin{aligned}
\hat{H}_{w_s} &= [R_{11} - \hat{c}^{(11)} e^{(11)^T}, \ldots, R_{1q_1} - \hat{c}^{(1q_1)} e^{(1q_1)^T}, \\
&\quad R_{21} - \hat{c}^{(21)} e^{(21)^T}, \ldots, R_{2q_2} - \hat{c}^{(2q_2)} e^{(2q_2)^T}, \\
&\quad \ldots, R_{p1} - \hat{c}^{(p1)} e^{(p1)^T}, \ldots, R_{pq_p} - \hat{c}^{(pq_p)} e^{(pq_p)^T}] \in \mathbb{R}^{n \times n}, \qquad (40) \\
\hat{H}_{b_s} &= [\sqrt{n_{11}}(\hat{c}^{(11)} - \hat{c}^{(1)}), \ldots, \sqrt{n_{1q_1}}(\hat{c}^{(1q_1)} - \hat{c}^{(1)}), \\
&\quad \sqrt{n_{21}}(\hat{c}^{(21)} - \hat{c}^{(2)}), \ldots, \sqrt{n_{2q_2}}(\hat{c}^{(2q_2)} - \hat{c}^{(2)}), \\
&\quad \ldots, \sqrt{n_{p1}}(\hat{c}^{(p1)} - \hat{c}^{(p)}), \ldots, \sqrt{n_{pq_p}}(\hat{c}^{(pq_p)} - \hat{c}^{(p)})] \in \mathbb{R}^{n \times s} \qquad (41) \\
\hat{H}_w^h &= [\sqrt{\alpha}\hat{H}_{w_s} \ \sqrt{1-\alpha}\hat{H}_{b_s}], \text{ and} \\
\hat{H}_b &= [\sqrt{n_1}(\hat{c}^{(1)} - \hat{c}), \sqrt{n_2}(\hat{c}^{(2)} - \hat{c}), \ldots, \sqrt{n_k}(\hat{c}^{(k)} - \hat{c})] \in \mathbb{R}^{n \times k}, \qquad (42)
\end{aligned}$$

where $\hat{c}^{(ij)} = Q_1^T c^{(ij)} \in \mathbb{R}^{n \times 1}$, $\hat{c}^{(i)} = Q_1^T c^{(i)} \in \mathbb{R}^{n \times 1}$, $\hat{c} = Q_1^T c \in \mathbb{R}^{n \times 1}$, and $s = \sum_{i=1}^{p} q_i$. It is easy to see that

$$\hat{H}_{w_s} = Q_1^T H_{w_s}, \ \hat{H}_{b_s} = Q_1^T H_{b_s} \ , \ \hat{H}_w^h = Q_1^T H_w^h \text{ and } \hat{H}_b = Q_1^T H_b.$$

Then the scatter matrices can be represented as

$$\hat{S}_{w_s} = \hat{H}_{w_s}\hat{H}_{w_s}^T = Q_1^T H_{w_s} H_{w_s}^T Q_1 = Q_1^T S_{w_s} Q_1, \qquad (43)$$

$$\hat{S}_{b_s} = \hat{H}_b \hat{H}_{b_s}^T = Q_1^T H_{b_s} H_{b_s}^T Q_1 = Q_1^T S_{b_s} Q_1, \qquad (44)$$

$$\hat{S}_w^h = \hat{H}_w^h (\hat{H}_w^h)^T = \alpha \hat{S}_{w_s} + (1-\alpha)\hat{S}_{b_s}, \text{ and} \qquad (45)$$

$$\hat{S}_b = \hat{H}_b \hat{H}_b^T = Q_1^T H_b H_b^T Q_1 = Q_1^T S_b Q_1. \qquad (46)$$

Suppose we find a matrix $\hat{G}$ that minimizes $trace(\hat{G}^T \hat{S}_w^h \hat{G})$ and maximizes $trace(\hat{G}^T \hat{S}_b \hat{G})$. Since

$$\begin{aligned}
\hat{G}^T \hat{S}_w^h \hat{G} &= \hat{G}^T(\alpha \hat{S}_{w_s} + (1-\alpha)\hat{S}_{b_s})\hat{G} \\
&= \hat{G}^T(\alpha \hat{H}_{w_s}\hat{H}_{w_s}^T + (1-\alpha)\hat{H}_{b_s}\hat{H}_{b_s}^T)\hat{G} \qquad (47) \\
&= \hat{G}^T Q_1^T(\alpha H_{w_s} H_{w_s}^T + (1-\alpha) H_{b_s} H_{b_s}^T) Q_1 \hat{G}, \text{ and} \qquad (48) \\
\hat{G}^T \hat{S}_b \hat{G} &= \hat{G}^T \hat{H}_b \hat{H}_b^T \hat{G} = \hat{G}^T Q_1^T H_b H_b^T Q_1 \hat{G} = \hat{G}^T Q_1^T S_b Q_1 \hat{G}, \qquad (49)
\end{aligned}$$

$G = Q_1\hat{G}$ provides the solution for minimizing $trace(G^T S_w^h G)$ and maximizing $trace(G^T S_b G)$, which is the h-LDA/GSVD solution shown in Algorithm 2. The above discussion shows that preprocessing by the QR decomposition can result in more efficiency in solving h-LDA/GSVD since it allows us to manipulate matrices of much smaller size, i.e. $n \times n$ rather than $m \times m$, when the dimension m of the data is very high while there are not as many data items, i.e. n with $m \gg n$.

We now show that regularization on $\hat{S_w^h}$ with the term γI_n is equivalent to regularization on S_w^h with the term γI_m. Consider

$$\max_{\hat{G}} trace((\hat{G}^T(\hat{S_w^h} + \gamma I_n)\hat{G})^{-1}(\hat{G}^T\hat{S}_b\hat{G})) \text{ and} \tag{50}$$

$$\max_{G} trace((G^T(S_w^h + \gamma I_m)G)^{-1}(G^T S_b G)). \tag{51}$$

Note that

$$\hat{S_w^h} + \gamma I_n = \begin{pmatrix} \hat{H_w^h} & \sqrt{\gamma} I_n \end{pmatrix} \begin{pmatrix} (\hat{H_w^h})^T \\ \sqrt{\gamma} I_n \end{pmatrix}.$$

Since $\hat{H_w^h}(\hat{H_w^h})^T = Q_1^T H_w^h (H_w^h)^T Q_1$ and $\gamma I_n = \gamma Q_1^T Q_1$, we have

$$\hat{S_w^h} + \gamma I_n = \hat{H_w^h}(\hat{H_w^h})^T + \gamma I_n = Q_1^T(H_w^h(H_w^h)^T + \gamma I_m)Q_1 = Q_1^T(S_w^h + \gamma I_m)Q_1. \tag{52}$$

From Eqs. (49-50) and (52), we obtain

$$\begin{aligned} \max_{\hat{G}} trace((\hat{G}^T(\hat{S_w^h} + \gamma I_n)\hat{G})^{-1}(\hat{G}^T\hat{S}_b\hat{G})) &= \max_{\hat{G}} trace((\hat{G}^T Q_1^T(S_w^h + \gamma I_m)Q_1\hat{G})^{-1}(\hat{G}^T Q_1^T S_b Q_1\hat{G})) \\ &= \max_{G} trace((G^T(S_w^h + \gamma I_m)G)^{-1}(G^T S_b G)), \end{aligned} \tag{53}$$

where $G = Q_1\hat{G}$. Eq. (53) shows that the solution obtained from regularization, after QR preprocessing, is equivalent to the LDA with regularization applied to the full space.

By applying QR preprocessing followed by regularization to Algorithm 2, we develop the algorithm for h-RLDA, which is summarized in Algorithm 3.

VI. EXPERIMENTS

A. Experimental Setup

In order to study the practical advantages of h-RLDA, we have applied it to a face recognition problem using Shimon Edelman's face database[1]. This data set contains 28 images that vary

[1] ftp://ftp.wisdom.weizmann.ac.il/pub/facebase

Algorithm 3 h-RLDA/QR-GSVD

Given a data matrix $A \in \mathbb{R}^{m\times n}$ with $m \geq n$ where the columns are partitioned into p clusters, and each cluster is further clustered into q_i clusters for $i = 1, \ldots, p$, respectively, and a regularization parameter $\gamma > 0$, this algorithm computes the dimension reducing transformation $G \in \mathbb{R}^{m\times(p-1)}$. For any vector $x \in \mathbb{R}^{m\times 1}$, $y = G^T x \in \mathbb{R}^{(p-1)\times 1}$ gives a $(p-1)$ dimensional representation of x.

1) Compute the reduced QR decomposition of A, i.e.,

$$A = Q_1 R$$

where $Q_1 \in \mathbb{R}^{m\times n}$ has orthonormal columns and $R \in \mathbb{R}^{n\times n}$ is upper triangular.

2) Compute $\hat{H}_{w_s} \in \mathbb{R}^{n\times n}$, $\hat{H}_{b_s} \in \mathbb{R}^{n\times s}$, and $\hat{H}_b \in \mathbb{R}^{n\times p}$ from R according to Eqs. (40), (41), and (42) respectively, where $s = \sum_{i=1}^{p} q_i$.

3) Compute the reduced QR decompositon of $\hat{K}_\gamma = \begin{pmatrix} \hat{H}_b^T \\ \sqrt{\alpha}\hat{H}_{w_s}^T \\ \sqrt{1-\alpha}\hat{H}_{b_s}^T \\ \sqrt{\gamma} I_n \end{pmatrix} \in \mathbb{R}^{(p+2n+s)\times n}$, i.e., $\hat{P}_\gamma^T \hat{K}_\gamma = \hat{R}_\gamma$, where $\hat{P}_\gamma \in \mathbb{R}^{(p+2n+s)\times n}$ has orthonormal columns and $\hat{R}_\gamma \in \mathbb{R}^{n\times n}$ is upper triangular.

4) Compute $\hat{W}_\gamma$ from the SVD of $\hat{P}_\gamma(1:p, 1:n)$, i.e., $\hat{U}_\gamma^T \hat{P}_\gamma(1:p, 1:n)\hat{W}_\gamma = \hat{\Sigma}_\gamma$.

5) Solve the triangular system $\hat{R}_\gamma \hat{G}_\gamma = \hat{W}_\gamma(:, 1:p-1)$ for $\hat{G}_\gamma$.

6) $G = Q_1 \hat{G}_\gamma$.

depending on such factors as angles of view, various illuminations, and facial expressions. We resized the original 512×352 pixel images to 64×44 pixel images due to memory restriction in computation, and preprocessed them using the contrast-limited adaptive histogram equalization scheme [22]. Then each 2-dimensional image is represented as a long column vector by stacking up the columns. Each image is given a set of labels that contain person id, angle of view, illumination, and facial expression.

Table I shows the data sets for classification experiments. In Data 1, 2, and 3, id was set to the target label to classify as in general face recognition applications, and one of the other labels was used as the subcluster label for h-RLDA. In Data 4, 5, and 6, labels other than id are used

as the target label under different subcluster labels. Discriminating labels other than id can be taken advantage of in various fields of study. For example, detecting facial expression is useful in psychology [23].

The proposed h-RLDA is compared to PCA, LDA/GSVD [1], and TensorFaces [10] as a preprocessing step for dimension reduction in classification.

PCA does not use any label information while the other methods do. LDA/GSVD uses only the person id, but h-RLDA and TensorFaces utilize subcluster information <?>in addition to<?> person id. For TensorFaces method, first, multilinear tensor data is constructed using all the available label information. For example, if the data is composed of 10 persons' 20×40 pixel images with variations of 5 angles, 3 illuminations, and 4 facial expressions, the size of the data tensor D would be $10 \times 5 \times 3 \times 4 \times 800$. Then, Higher-Order SVD (HOSVD) [24] is performed on the tensor data D. For every possible combination of angles, illuminations, and expressions, the dimension reducing matrices of size 10×800 are obtained from the computed core tensor of HOSVD.

Dimension reducing matrices are then applied to the original data, and K-nearest neighbor classification, where $K = 1$, is performed to estimate the label of each test data. In the case of TensorFaces, since it produces multiple dimension reducing matrices, the training and test images are mapped using each of them, and the closest training image to the target test image is chosen as a candidate in each subspace. Among those closest training images, one that has the minimum distance is chosen, and its label is assigned to the test image.

As a performance measure, we present recognition accuracies and computation time required for each method. For h-RLDA, the parameters α ($0 \leq \alpha \leq 1$) and γ in Eq. (38) were optimized using k-fold cross-validation with step size of 0.1 for α, and $2^{-i},\ i = 1, 2, \cdots, 30$ for γ, respectively. In the case of multiple pairs of values of α and γ produced the best cross-validation accuracy, we chose the smallest value for γ and then the largest for α.

All the experiments were done using Matlab on Windows XP with 1.6GHz CPU with 1.5GB memory, and the Matlab Tensor Toolbox[2] was used for TensorFaces algorithm.

[2]http://csmr.ca.sandia.gov/~tgkolda/TensorToolbox/

TABLE I

DESCRIPTION OF DATA SETUP

	Target Label	Subcluster Label	Training/Test Data	#data	#dim
Data 1	Person	Angle of View	Training : **3 angles of view**($0^\circ, \pm 34^\circ$), 3 illuminations, 3 facial expressions	729	2816
			Test : **2 other angles of view**($\pm 17^\circ$), 3 illuminations, 3 facial expressions	486	
Data 2		Illumination	Training : 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), **2 illuminations**, 3 facial expressions	810	
			Test : 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), **1s other illuminations**, 3 facial expressions	405	
Data 3		Facial Expression	Training : 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), 3 illuminations, **2 facial expressions**	810	
			Test : 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), 3 illuminations, **1 other facial expression**	405	
Data 4	Angle of View	Person	Training : **8 persons**, 3 illuminations, 3 facial expressions	360	
			Test : **19 other persons**, 3 illuminations, 3 facial expressions	855	
Data 5	Illumination		Training : **8 persons**, 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), 3 facial expressions	360	
			Test : **19 other persons**, 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), 3 facial expressions	855	
Data 6	Facial Expression		Training : **8 persons**, 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), 3 illuminations	360	
			Test : **19 other persons**, 5 angles of view($0^\circ, \pm 17^\circ \pm 34^\circ$), 3 illuminations	855	

B. Results

For six different training/test sets, the recognition accuracies of PCA, LDA, TensorFaces, and h-RLDA are shown in Table II. Theoretically, the maximum possible reduced dimension is n, $p-1$, p, and $p-1$ for PCA, LDA, TensorFaces, and h-RLDA, respectively, where n is the number of training data, and p is the number of clusters to classify. For fair comparison in terms of reduced dimension, the results from p leading eigenvectors and two intermediate dimensions for PCA are also presented in Table II. In all cases, h-RLDA shows consistently better performance. Another interesting observation is that TensorFaces did not outperform PCA as clearly as reported in [10], and it did not perform as well as LDA except for Data 4 although it utilized more information than LDA. Although the difference of accuracies between LDA and h-RLDA was not significant with Data 1, 2, and 3, it was significant with Data 4, 5, and 6. In general, factors such as illumination and facial expression may be harder to classify than person id. They would have distinct subcluster structures depending on, say, person id, and some of the subclusters may be rather close to the data with other target labels. For instance, an image with a smiling facial expression of person #1 may be much closer to that with a frowning expression of the same person rather than the smiling face image of person #2 (see Figure 4). In this case, making the distance among the data points in subclusters within a cluster shorter may also bring

TABLE II

COMPARISON OF FACE RECOGNITION ACCURACIES(%)

		PCA				LDA	TensorFaces	h-RLDA
Data 1	Dimension	729	300	100	27	26	27	26
	Accuracy	88.73%	89.12%	87.75%	81.98%	96.24%	85.92%	**98.59%**
Data 2	Dimension	810	300	100	27	26	27	26
	Accuracy	86.47%	85.83%	85.83%	80.47%	97.58%	90.94%	**99.82%**
Data 3	Dimension	810	300	100	27	26	27	26
	Accuracy	87.15%	87.15%	88.64%	83.45%	98.34%	89.42%	**100%**
Data 4	Dimension	360	150	50	5	4	5	4
	Accuracy	89.27%	88.89%	86.12%	79.83%	89.27%	93.63%	**95.53%**
Data 5	Dimension	360	150	50	3	2	3	2
	Accuracy	77.72%	78.26%	75.84%	70.29%	80.15%	64.72%	**91.24%**
Data 6	Dimension	360	150	50	3	2	3	2
	Accuracy	64.19%	64.91%	63.53%	60.42%	75.83%	69.61%	**81.95%**

TABLE III

COMPARISON OF COMPUTATION TIMES IN SECONDS REQUIRED TO RUN DATA 1 IN TABLE I

	PCA	LDA/ GSVD	TensorFaces	h-LDA/ GSVD	h-RLDA/ QR-regGSVD
Generating dimension reducing matrices or tensors	2.86	25.14	703	32.79	4.15
Performing classification using 1-NN	1.33	0.12	1120	0.12	0.12
Total computation time	4.19	25.26	1823	32.91	4.27

data points in nearby subclusters from other clusters together, keeping LDA from separating different clusters in the reduced space. On the other hand, h-RLDA can handle such cases by adjusting the weights of the within-subcluster and the between-subcluster scatter matrices, which explains the clearer difference in prediction accuracies between LDA and h-RLDA in Data 4, 5, and 6.

In addition, Figure 3 shows the recognition accuracies for various reduced dimensions in the case of Data 1, where the reduced dimension can be for less than the theoretically optimal reduced dimension. In PCA, the reduced space of dimension $d \leq (p-1)$ was obtained from the d leading eigenvectors, and in LDA and h-RLDA, the d leading generalized singular vectors.

Fig. 3. Recognition accuracies versus subspace dimensionality of Data 1 shown in Table I

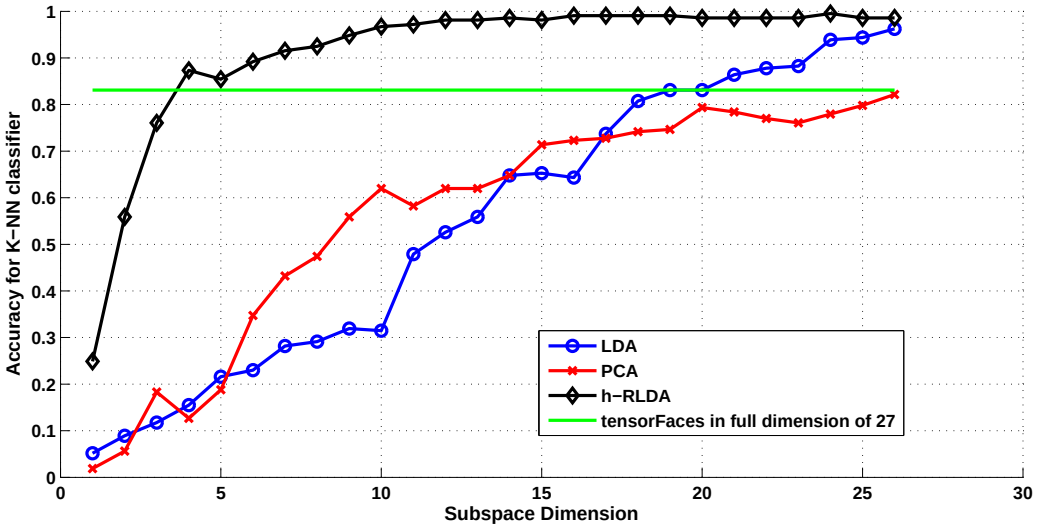
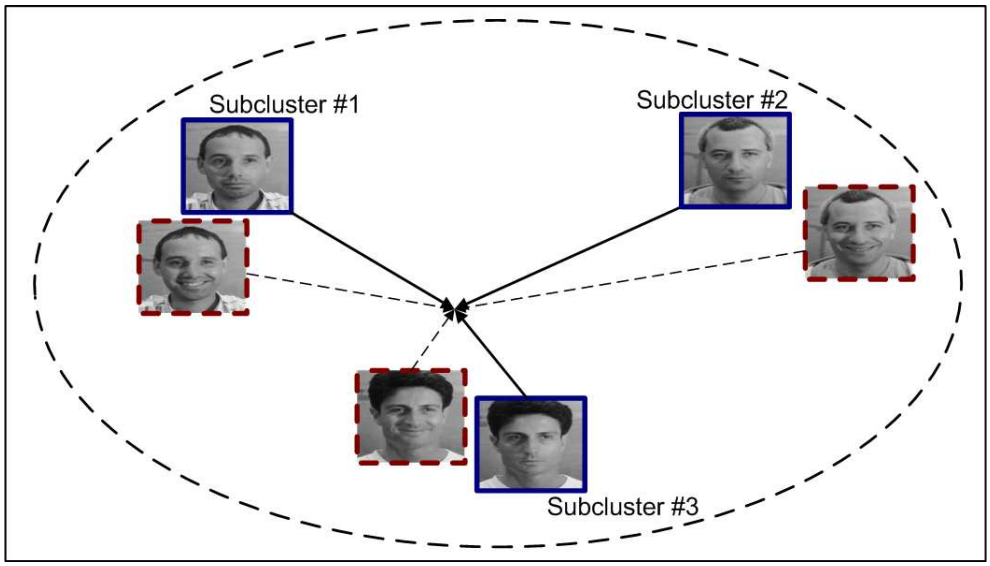


Fig. 4. Example of in facial expression recognition. Images with the dotted line belong to cluster #1 (smiling facial expression) and images with the solid line belong to cluster #2 (no facial expression) which can be further clustered into three subclusters depending on person id. Note that images of two different clusters are located closely to each other according to person id. When minimizing within-cluster distances in cluster #2, LDA may be interfered with the images in cluster #1.



For TensorFaces, since the concept of leading basis vectors in HOSVD is not clear, we only present the accuracy in full dimensionality. From Figure 3, we can observe that h-RLDA reaches its maximum performance very fast even with the reduced dimension of about 10 whereas LDA requires almost the the theoretically optimal reduced dimension $p-1$ to produce its best performance. This shows that the quality of the extracted features of h-RLDA can be much better than that of LDA especially when the reduced dimension is significantly smaller than the

theoretically optimal dimension of $p-1$. Such advantages would also be exploited in applications such as visualization of cluster structures where substantial dimension reduction to the reduced dimension of 2 or 3 is required.

Finally, Table III shows the computation time required to run each method. From this table, h-RLDA/QR-GSVD proves its efficiency due to QR preprocessing over LDA/GSVD and h-LDA/GSVD, and h-RLDA is much faster than TensorFaces, while both utilize additional label information other than person id. The reason why 1-NN performs very slow in TensorFaces is because mapping each test image into derived subspaces involves linear system solving instead of simple matrix-vector multiplication [10].

VII. Conclusions

In this paper, a novel concept of hierarchical LDA (h-LDA) is introduced by deriving the within-cluster scatter matrices using additional information available allowing clustering the data further with subclusters. The new h-LDA generalized the applicability of LDA to the cases when the assumption of a unimodal Gaussian model in each cluster is not necessarily valid. We also presented its theoretical relationship to two-way MANOVA in the context of hypothesis testing. Utilizing regularization and adopting the regularized LDA algorithm with QR decomposition preprocessing, an efficient regularized h-LDA (h-RLDA) algorithm is designed and hierarchical LDA was successfully applied to the face recognition problem, and demonstrates superior performance over other methods such as PCA, LDA, and TensorFaces in terms of prediction accuracy as well as computational complexity.

References

[1] P. Howland, J. Wang, and H. Park, "Solving the small sample size problem in face recognition using generalized discriminant analysis," *Journal of Pattern Recogntion*, vol. 39, no. 2, pp. 277–287, 2006.

[2] D. L. Swets and J. J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831–836, 1996.

[3] P. N. Belhumeur, J. P. Hespanha, and D. J. 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.

[4] P. Howland and H. Park, "Equivalence of Several Two-stage Methods for Linear Discriminant Analysis," *Proceedings of the Fourth SIAM International Conference on Data Mining*, pp. 69–77, 2004.

[5] P. Howland, M. Jeon, and H. Park, "Structure preserving dimension reduction for clustered text data based on the generalized singular value decomposition," *SIAM Journal on Matrix Analysis and Applications*, vol. 25, no. 1, pp. 165–179, 2003.

[6] H. Park, B. Drake, S. Lee, and C. Park, "Fast Linear Discriminant Analysis using QR Decomposition and Regularization," *Technical Report GT-CSE-07-21*, 2007.

[7] K. Fukunaga and J. M. Mantock, "Nonparametric discriminant analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 671–678, 1983.

[8] M. Zhu and A. Martinez, "Subclass Discriminant Analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1274–1286, 2006.

[9] J. Friedman, "Regularized Discriminant Analysis," *Journal of the American Statistical Association*, vol. 84, no. 405, pp. 165–175, 1989.

[10] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear image analysis for facial recognition," in *Proceedings of International Conference on Pattern Recognition (ICPR 2002)*, (Quebec City, Canada), pp. 511–514, August.

[11] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York: Wiley-interscience, 2001.

[12] C. Bishop, *Pattern recognition and machine learning*. Springer, 2006.

[13] A. Jain and R. Dubes, *Algorithms for clustering data*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1988.

[14] K. Fukunaga, *Introduction to Statistical Pattern Recognition, second edition*. Boston: Academic Press, 1990.

[15] G. H. Golub and C. F. van Loan, *Matrix Computations, third edition*. Johns Hopkins University Press, Baltimore, 1996.

[16] C. C. Paige and M. A. Saunders, "Towards a generalized singular value decomposition," *SIAM J. Numer. Anal.*, vol. 18, pp. 398–405, 1981.

[17] C. F. van Loan, "Generalizing the singular value decomposition," *SIAM J. Num. Anal.*, vol. 13, pp. 76–83, 1976.

[18] P. Howland and H. Park, "Generalizing discriminant analysis using the generalized singular value decomposition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 8, pp. 995–1006, 2004.

[19] R. Johnson and D. Wichern, *Applied multivariate statistical analysis*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1988.

[20] T. Hastie, R. Tibshirani, and J. Friedman, "The Elements of Statistical Learning," 2001.

[21] W. Krzanowski, *Principles of Multivariate Analysis: A User's Perspective*. Oxford University Press, 2000.

[22] A. Jain, *Fundamentals of digital image processing*. Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 1989.

[23] M. Pantic and L. J. Rothkrantz, "Automatic analysis of facial expressions: The state of the art," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1424–1445, 2000.

[24] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*. to appear (accepted June 2008).