Manifold Discriminant Analysis

Ruiping Wang[1, 2, 3], Xilin Chen[1, 2]

[1]Key Lab of Intelligent Information Processing, Chinese Academy of Sciences (CAS), Beijing, China
[2]Institute of Computing Technology, CAS, Beijing, 100190, China
[3]Graduate University of Chinese Academy of Sciences, Beijing, 100049, China
{rpwang, xlchen}@jdl.ac.cn

Abstract

This paper presents a novel discriminative learning method, called Manifold Discriminant Analysis (MDA), to solve the problem of image set classification. By modeling each image set as a manifold, we formulate the problem as classification-oriented multi-manifolds learning. Aiming at maximizing "manifold margin", MDA seeks to learn an embedding space, where manifolds with different class labels are better separated, and local data compactness within each manifold is enhanced. As a result, new testing manifold can be more reliably classified in the learned embedding space. The proposed method is evaluated on the tasks of object recognition with image sets, including face recognition and object categorization. Comprehensive comparisons and extensive experiments demonstrate the effectiveness of our method.

1. Introduction

Manifold learning has been an active topic in the community of computer vision and pattern recognition for many years. Classical methods such as Isomap [24], LLE [20], and Laplacian Eigenmap [3] mainly address data representation and all seek to model a single manifold in an unsupervised manner. In addition, due to the implicitness of their nonlinear maps, they can not be directly applied to new test samples. Such properties limit the application of these methods to classification tasks. Recently, several algorithms have been developed to provide mappings for the whole data space, e.g., Locality Preserving Projections (LPP) [12], Unsupervised Discriminant Projection (UDP) [30]. While their mappings are linear and easily computable, they have shown encouraging results on biometric tasks. Yet, both methods are still essentially unsupervised and may not promise good discriminating capability in some cases [7].

For classification tasks, Linear Discriminant Analysis (LDA) has proved its effectiveness on various applications. However, as a parametric method, LDA highly depends on the data distribution and can only deal with linearly separable problems well. To improve LDA, many extensions have been developed, such as Nonparametric Discriminant Analysis (NDA) [5], Subclass Discriminant Analysis (SDA) [33], Maximum Margin Criterion (MMC) [18], and Kernel Fisher Discriminant (KFD) [19].

More recently, from the perspective of manifold learning, a few algorithms were proposed to address classification problems, e.g., Marginal Fisher Analysis (MFA) [28], Local Discriminant Embedding (LDE) [7], Laplacian SVM (LapSVM) [4], and some others [22], [29]. While these methods have demonstrated good results in traditional classification tasks, their learning and testing are still based on single sample, just like the manner of LDA. They rarely investigate the property of the whole manifold explicitly. In other words, they are not specifically designed for the task of set classification. In this paper, we will address this issue and verify it by the problem of Object Recognition with Image Sets (ORIS) [26].

1.1. Previous work

In the task of ORIS, each set contains images of the same class covering large appearance variations. Recognition is carried out by classifying an unknown set of images to one of the training classes, each also represented by an image set. Since the information of image sets can be efficiently exploited, more robust recognition performance can be expected by using set as input rather than single image [15], [26]. Whereas much previous work on matching image sets for object recognition exploits video dynamics information [16], [23], [32], this paper does not make such assumption, and thus has less relation to those methods.

From the view of set modeling, relevant approaches to set classification broadly fall into two classes: model-based parametric methods and model-free nonparametric methods. Typical model-based methods [1], [21], tend to represent image set by parametric distribution function, and then measure the similarity between two distributions. These methods often suffer from the difficulty of parameter estimation, and may not work well when the training and novel test data sets have weak statistical correlations [15]. In contrast, model-free nonparametric methods attempt to represent the image set either by linear subspace [15], [27] or by nonlinear manifold [9], [11], [26], [31]. Without any assumptions on data distribution, nonparametric methods thus come with many favorable properties and attract more and more attentions in the field.

The above model-free nonparametric methods can also be considered from the viewpoint of classification purpose. Generally, some methods purely concern with how to measure the similarity of two sets [11], [26], [27]. On the

978-1-4244-3991-1/09/$25.00 ©2009 IEEE

contrary, others pay more attention to the problem of learning discriminant function from training data with a given similarity function [9], [15], [31].

1.2. Our approach

The recent work [26] proposed to model image set as manifold and formulated the ORIS task as the computation of Manifold-Manifold Distance (MMD). However, no discriminative information is exploited in [26], which is not attractive enough for classification. Having this in mind, in this paper we propose a novel discriminative learning method, called Manifold Discriminant Analysis (MDA). MDA aims to maximize the "margin" of manifolds with different class labels while enhancing the local data compactness within each manifold.

The key points of MDA lie in two folds: 1) Local linear model. For each manifold, an effective clustering method is conducted to extract a set of clusters, with each cluster being one local linear model. The margin between two global nonlinear manifolds is then characterized by that between their corresponding local linear models. 2) Discriminative learning. Inspired by LDA and MMC [18], MDA seeks to learn a linear discriminant function to map the multi-class manifolds into an embedding space. In the embedding, local models from the manifolds with different class labels can be better separated, and meanwhile neighbor relationships within each local model are preserved to maintain local data compactness. As a result, new testing manifold transformed by the discriminant function can be more reliably classified in the MDA embedding space.

In what follows, we present our algorithm in section 2. After a discussion on the relations between our MDA and other methods in section 3, we give extensive experimental evaluation in section 4. Conclusion is drawn in section 5.

2. Manifold Discriminant Analysis

In this section, we first give a primary formulation of the image set classification problem. Then, we describe the proposed MDA algorithm by emphasizing its two key points, i.e., local linear model, and discriminative learning.

2.1. Problem formulation

In view of the foregoing discussions, we model the image set as manifold, and formulate the problem of object recognition with image sets as classification-oriented multi-manifolds learning [30]. Formally, assume we are given M image sets as: $\{X_1, X_2, ..., X_M\}$, where $X_i = [\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, ..., \boldsymbol{x}_{i,Ni}]$ ($i = 1, ..., M$) describes a data matrix of the i-th set, and Ni denotes the number of image samples. Each set belongs to one of object classes denoted by $\{L_i \mid L_i \in \{1, 2, ..., \mathrm{P}\}\}_{i=1}^{M}$. As stated above, we represent

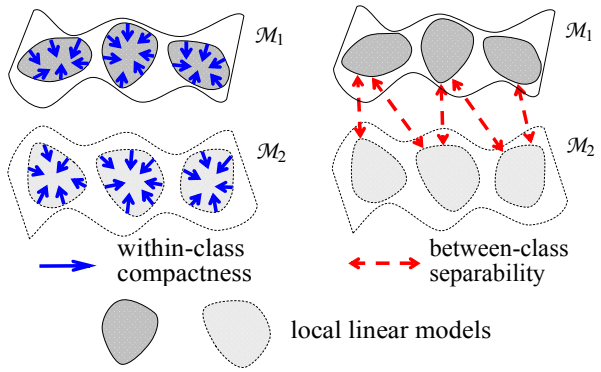


Figure 1: Conceptual illustration of the intrinsic graph (*left*) and penalty graph (*right*) for the proposed MDA. $\mathcal{M}_1$ and $\mathcal{M}_2$ are two manifolds with different class labels. In the intrinsic graph, data samples within each local linear model are to be compressed to enhance within-class compactness (see the *blue solid arrows*). In the penalty graph, neighboring local models which come from different classes' manifolds are to be pushed far away to reflect between-class separability (see the *red dashed arrows*).

each image set X_i as a nonlinear manifold $\mathcal{M}_i$, and then use a collection of local linear models to characterize it, e.g., $\mathcal{M}_i = \{C_{i,1}, C_{i,2}, \ldots, C_{i,ni}\}$. Here, ni denotes the number of local linear models in the i-th set. In most cases, ni is far smaller than Ni.

With the above assumption, the multi-manifolds learning problem transforms to learning a discriminating embedding space which can better distinguish different manifolds. In other words, we aim to maximize the "margin" of manifolds with different class labels. To our knowledge, the concept of "manifold margin" has not been well established. Nonetheless, with the local linear model representation in this study, we instead convert the margin between two global nonlinear manifolds to that between their corresponding local linear models.

Intuitively, the proposed MDA seeks to learn a linear discriminant function that maximizes the between-class manifolds separability and enhances the within-class local data compactness meanwhile. Motivated by the concept of "intrinsic graph" and "penalty graph" in [28], here we specifically design two such graphs to characterize the within-class compactness and between-class separability respectively, as shown in Fig. 1. Detailed description will be presented in the next sections.

2.2. Constructing local linear models

The idea of extracting local models from manifold has been exploited in several methods. They often adopt typical clustering methods, e.g., k-means [11], [16] or hierarchical agglomerative clustering (HAC) [9], [31], in a simple manner. As a result, their local models, i.e., clusters, do not have explicit linearity guarantee.

In our earlier work [26], a novel local linear model called maximal linear patch (MLP), was introduced directly from the view of manifold. In brief, MLP is defined as local linear patch on manifold, whose nonlinearity degree is measured by the deviation between Euclidean distances and geodesic distances [24]. To construct MLPs, we also developed a one-shot sequential clustering method. However, as we indicated in [26], this clustering method can suffer from the problem of unbalanced clusters, i.e., the clusters computed earlier may have much larger size than those ones obtained later. Also, the one-shot clustering could only yield MLPs for some given nonlinearity degree. If one wants MLPs for a different nonlinearity degree, the clustering process has to be conducted once again.

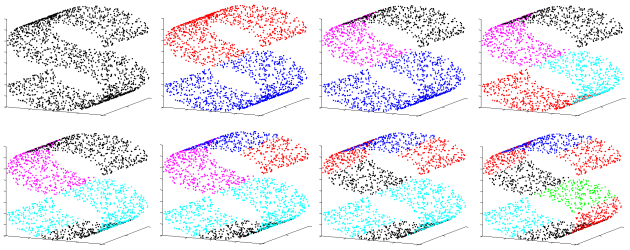
In this paper, we propose to combine the merits of MLP and hierarchical clustering method in a more effective and flexible way, since hierarchical clustering allows one to cluster data over different degrees by creating a cluster tree called *dendrogram*. In this study, however, we do not employ the HAC manner as previous work [9], [31]; instead, we explore Hierarchical Divisive Clustering (HDC) [13]. This is because, in most cases, the appropriate number of clusters is much smaller than the number of data samples, and we do not need a complete hierarchy all the way down to individual samples. Thus, the bottom-up HAC is much less efficient than the top-down HDC for our purpose.

The basic idea of our HDC algorithm is that, in the first level, all samples are initiated as a singleton MLP (cluster). Then, in each new level, the MLP in the previous level with the largest nonlinearity degree will split into two smaller ones, which are consequently with decreased nonlinearity degrees. Finally, we are able to obtain multi-level local models associated with different nonlinearity degrees.

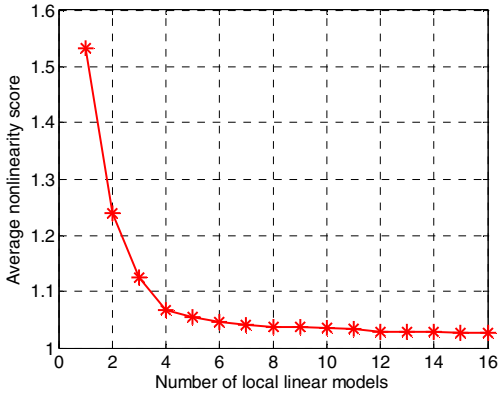
Following the notations of Sec.2.1, we give the detailed implementation of the HDC algorithm. For a given manifold $\mathcal{M}_i$ with its data set $X_i = [\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, ..., \boldsymbol{x}_{i,Ni}]$, we aim to extract a collection of MLPs, i.e., local models $C_{i,k}$,

$$\mathcal{M}_i = \{C_{i,1}, C_{i,2}, \ldots, C_{i,ni}\}, \text{ where}$$
$$C_{i,k} \,|_{k=1}^{ni} = \{\boldsymbol{x}_{i,1}^{(k)}, \boldsymbol{x}_{i,2}^{(k)}, ..., \boldsymbol{x}_{i,a_k}^{(k)}\}, \ (\sum_{k=1}^{ni} a_k = Ni). \quad (1)$$

Firstly, the pair-wise Euclidean distance matrix D_E and geodesic distance matrix D_G (based on k-NN graph) are computed. Then a matrix holding distance ratios is obtained as: $R(\boldsymbol{x}_m, \boldsymbol{x}_n) = D_G(\boldsymbol{x}_m, \boldsymbol{x}_n) / D_E(\boldsymbol{x}_m, \boldsymbol{x}_n)$. Clearly, these three matrices are all of size $Ni \times Ni$. Since geodesic distance is always no smaller than Euclidean distance, $R(\boldsymbol{x}_m, \boldsymbol{x}_n) \geq 1$ holds for any entry of R. Besides, another matrix H, of size $k \times Ni$, is also constructed, each column $H(:,l)$ $(l = 1, ..., Ni)$ holding the k-NNs' indices of the point $\boldsymbol{x}_{i,l}$. To measure the nonlinearity degree of one local model (MLP) $C_{i,k}$, we can define a *nonlinearity score function* as:



(a) HDC clustering dendrogram



(b) Average nonlinearity score in each clustering level

Figure 2: Clustering results of our HDC algorithm on the classical "S-curve" manifold, which has 2000 uniformly sampled data points. See text below for details. For better viewing, please see the color PDF file.

$$S_{i,k} = \frac{1}{a_k \cdot a_k} \sum_{m=1}^{a_k} \sum_{n=1}^{a_k} R(\boldsymbol{x}_{i,m}^{(k)}, \boldsymbol{x}_{i,n}^{(k)}) \,. \quad (2)$$

With these definitions, our HDC algorithm is formulated below as *Algo.1* in Table 1. Note that the *threshold* δ in step.2 controls the termination of the algorithm, and thus the number of final clusters as well as their nonlinearity degrees. Obviously, the complete clustering hierarchy can be produced whenever δ is specified to any value less than 1. From Table 1, one can see that most steps of *Algo.1* are accessing operations against existing matrices computed in advance. Although it involves some iterative steps, the algorithm runs very efficiently nevertheless.

By applying *Algo.1* to the classical "S-curve" manifold, we can get the clustering dendrogram, with the first 8 levels illustrated in Fig.2(a). From the figure, one can find that the problem of unbalanced clusters of [26] has been effectively alleviated. The average nonlinearity score of all MLPs in each level is shown in Fig.2(b). As can be seen, the average nonlinearity score decreases as the levels and MLPs are increased. Fortunately, the curve in Fig.2(b) provides an easy guide to select the proper number of local linear models for subsequent uses. A simple but effective choice is the *elbow* of the curve, at which the curve ceases to decrease significantly with added local linear models.

Table 1. Clustering method for local linear model construction

Algorithm 1. Hierarchical Divisive Clustering (HDC) algorithm

Input: manifold $\mathcal{M}_i$ with its data set $X_i = [\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, ..., \boldsymbol{x}_{i,Ni}]$

Output: local linear model representation $\mathcal{M}_i = \{C_{i,1}, C_{i,2}, \ldots, C_{i,ni}\}$

1 Initialization: $C_{i,1} = \{\boldsymbol{x}_{i,1}, \boldsymbol{x}_{i,2}, ..., \boldsymbol{x}_{i,Ni}\}$, $ni = 1$; compute $S_{i,1}$ according to Eq.(2).

2 Choose $C_{i,k}$ ($k \in \{1, 2, ..., ni\}$) with the largest nonlinearity score $S_{i,k}$.

If $S_{i,k} \le \delta$ (δ is a *threshold*), return the current clustering results and HDC terminates; else, go to step 2.1.

2.1 According to geodesic distance matrix D_G, select two furthest seed points, $\boldsymbol{x}_L$ and $\boldsymbol{x}_R$, from $C_{i,k}$.

Initialize two new clusters: $C_{i,k}^{(L)} = \{\boldsymbol{x}_L\}$, $C_{i,k}^{(R)} = \{\boldsymbol{x}_R\}$. Update: $C_{i,k} \leftarrow C_{i,k} \setminus \{\boldsymbol{x}_L, \boldsymbol{x}_R\}$.

2.2 Update $C_{i,k}^{(L)}$ and $C_{i,k}^{(R)}$ by iteratively running step 2.2.1-2.2.2 until $C_{i,k} = \varnothing$:

2.2.1 For current $C_{i,k}^{(L)}$, construct its neighbor points set, denote by $P^{(L)}$. According to the matrix H, $P^{(L)}$ gathers the *k*-NN samples of all the points in $C_{i,k}^{(L)}$. In the same way, construct $P^{(R)}$ for $C_{i,k}^{(R)}$.

2.2.2 $C_{i,k}^{(L)} \leftarrow C_{i,k}^{(L)} \cup (P^{(L)} \cap C_{i,k})$, $C_{i,k} \leftarrow C_{i,k} \setminus (P^{(L)} \cap C_{i,k})$; $C_{i,k}^{(R)} \leftarrow C_{i,k}^{(R)} \cup (P^{(R)} \cap C_{i,k})$, $C_{i,k} \leftarrow C_{i,k} \setminus (P^{(R)} \cap C_{i,k})$.

3 The single cluster $C_{i,k}$ splits into two ones: $C_{i,k}^{(L)}$ and $C_{i,k}^{(R)}$.

Update: $ni \leftarrow ni + 1$, compute $S_{i,k}^{(L)}$ and $S_{i,k}^{(R)}$ according to Eq.(2). Go to step 2.

2.3. Learning discriminant function

Now let us return to the two graphs in Fig.1. To construct the intrinsic graph, samples from the same local model are connected to each other. For the penalty graph, we need to connect the neighboring local model pairs, which come from two manifolds with different class labels. Then how to measure the distance between a pair of local models?

Given two manifolds: $\mathcal{M}_i = \{C_{i,1}, C_{i,2}, \ldots, C_{i,ni}\}$ and $\mathcal{M}_j = \{C_{j,1}, C_{j,2}, \ldots, C_{j,nj}\}$, let $C_{i,k} \mid_{k=1}^{ni} = \{\boldsymbol{x}_{i,1}^{(k)}, \boldsymbol{x}_{i,2}^{(k)}, ..., \boldsymbol{x}_{i,a_k}^{(k)}\}$, $C_{j,l} \mid_{l=1}^{nj} = \{\boldsymbol{x}_{j,1}^{(l)}, \boldsymbol{x}_{j,2}^{(l)}, ..., \boldsymbol{x}_{j,a_l}^{(l)}\}$, we can simply define the distance between the local models by their sample centers:

$$d(C_{i,k}, C_{j,l}) = \left\| \boldsymbol{e}_{i,k} - \boldsymbol{e}_{j,l} \right\|, \text{ where}$$
$$\boldsymbol{e}_{i,k} = \frac{1}{a_k} \sum_{m=1}^{a_k} \boldsymbol{x}_{i,m}^{(k)}, \text{ and } \boldsymbol{e}_{j,l} = \frac{1}{a_l} \sum_{n=1}^{a_l} \boldsymbol{x}_{j,n}^{(l)}. \quad (3)$$

Next, writing the whole data matrix: $X = [X_1, X_2, ..., X_M] = [\boldsymbol{x}_{1,1}, \boldsymbol{x}_{1,2}, ..., \boldsymbol{x}_{1,N1}; \cdots\cdots; \boldsymbol{x}_{M,1}, \boldsymbol{x}_{M,2}, ..., \boldsymbol{x}_{M,NM}]$, for notation simplicity, we rewrite it as: $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N]$, where $\boldsymbol{x}_i \in \mathbb{R}^d$ ($i = 1, ..., N$), and $N = \sum_{i=1}^{M} Ni$ denotes the total number of training data samples. The learning method can be formulated as the following steps.

1. *Construct graphs*. Let G and G' refer to the intrinsic graph and penalty graph respectively (both over all data points). For each pair of points, $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$, denote their respective local models as $\boldsymbol{x}_m \in C_{i,k}$, $\boldsymbol{x}_n \in C_{j,l}$. To construct G, as stated above, we consider those point-pairs from the same local linear model, i.e., an edge is added between $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$ if $i = j$ and $k = l$. For G', we instead consider each pair of points from between-class neighboring local models, i.e., an edge is added between $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$, if $L_i \neq L_j$ (which are class labels) and $C_{i,k}$ is among the k'-nearest between-class neighbors of $C_{j,l}$ or vice versa. With this definition, if a training class has two or more manifolds, their neighboring local models will not be allowed to be connected in the penalty graph.

2. *Compute affinity matrices*. Denote the affinity matrix of G by W, whose element $w_{m,n}$ refers to the weight of edge between $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$. It can be defined as: $w_{m,n} = 1$ if $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$ are connected, and $w_{m,n} = 0$ otherwise. The other affinity matrix W' of G' is computed in the same way. It can be seen, both W and W' are $N \times N$ symmetric matrices. Here we use the "simple-minded" [3] affinity weights definition. Another possible choice is the "heat kernel" defined as, $w_{m,n} = \exp[-\left\| \boldsymbol{x}_m - \boldsymbol{x}_n \right\|^2 / t]$ if $\boldsymbol{x}_m$ and $\boldsymbol{x}_n$ are connected, and $w_{m,n} = 0$ otherwise.

3. *Compute the embedding space*. In our current study, MDA learns to construct the embedding $\boldsymbol{z} = V^T \boldsymbol{x}$ based on linear projections, where $V = [\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_l]$ is an $d \times l$ matrix with $l \ll d$. With the two affinity matrices above, we can then define two scatter terms S_w and S_b to characterize within-class compactness and between-class separability respectively as follows:

$$S_w = \sum_{m,n} \left\| \boldsymbol{v}^T \boldsymbol{x}_m - \boldsymbol{v}^T \boldsymbol{x}_n \right\|^2 w_{m,n} = 2\boldsymbol{v}^T X(D-W)X^T \boldsymbol{v} . \quad (4)$$

$$S_b = \sum_{m,n} \left\| \boldsymbol{v}^T \boldsymbol{x}_m - \boldsymbol{v}^T \boldsymbol{x}_n \right\|^2 w'_{m,n} = 2\boldsymbol{v}^T X(D'-W')X^T \boldsymbol{v} . \quad (5)$$

where D and D' are diagonal matrices with diagonal elements $d_{m,m} = \sum_n w_{m,n}$ and $d'_{m,m} = \sum_n w'_{m,n}$. It is well known that [8], $L_w = D - W$ and $L_b = D' - W'$, are the Laplacian matrices of two graphs G and G' respectively. Recall that, MDA aims to maximize the between-class separability and enhance the within-class compactness meanwhile. As a result, the objective function of MDA arrives at the following optimization criterion:

$$\text{Maximize } J(\boldsymbol{v}) = \frac{|S_b|}{|S_w|} = \frac{\boldsymbol{v}^T X L_b X^T \boldsymbol{v}}{\boldsymbol{v}^T X L_w X^T \boldsymbol{v}} \quad (6)$$

With some simple algebraic formulations, it can be seen that the columns of an optimal V are the generalized eigenvectors corresponding to the l largest eigenvalues in

$$XL_b X^T \boldsymbol{v} = \lambda XL_w X^T \boldsymbol{v} . \quad (7)$$

2.4. Classification by MDA

Once we have learned the projection matrix V, the classification of multi-manifolds can be conducted more reliably in the MDA embedding space. In this space, on one hand, the manifolds corresponding to different classes have been better disassociated; on the other hand, the data samples in the local models become more compact, thus the sample centers can better represent the local models.

Here we briefly describe the testing procedure of MDA. To compare two novel image sets, we first extract their local model representations by the HDC algorithm in Table 1. Then, pair-wise local model distances are computed in the learned MDA embedding space according to Eq. (3). Since the projection of MDA is linear, it only needs to compute the local model centers in the original data space, and then project the centers into the embedding space to compute distances. Finally, the similarity of two sets is obtained by matching their closest pair of local models. This also conforms to the suggestion in [15], [26] that, one of the most effective solutions to match two sets would be to measure the similarity of their most common parts of data. It can be seen that, the testing procedure of MDA is quite efficient.

3. Discussion

This section presents detailed comparisons between the proposed MDA and some well known related works.

MDA *vs.* MFA [28], LDE [7] and KFD [19]. The basic difference lies in that, MDA is a set-based classification method, while the others are sample-based, i.e., they rarely exploit the set information and are not specifically designed for the task of set classification. As noted in [30], MFA and LDE are essentially the same, and can be seen as localized and nonparametric variants of LDA. From the view of graph embedding framework [28], MDA shares the similar graphs motivation with these two methods in that, all focus on the characterization of within-class compactness and between-class separability. However, MDA mainly operates on the extracted manifold local models, both to construct the graphs for training and to measure the set similarity for testing, while all the operations of MFA and LDE are based on single samples. Concerning KFD [19], the most related part of it to our MDA is perhaps that both methods account for the nonlinearity of data distribution. However, as the nonlinear extension of LDA, KFD seeks to perform classification in some implicit higher-dimensional feature space by embedding the kernel trick. In contrast, MDA explicitly finds a lower-dimensional discriminative space to classify multiple nonlinear manifolds.

MDA *vs.* MMD [26] and DCC [15]. Both MMD and DCC (Discriminant-analysis of Canonical Correlations) are set-based methods which are directly applicable for image set classification. As stated in Sec.1.2, MMD proposed a framework to compute the distance of manifolds *without* mapping the data to some new space, while DCC tries to learn discriminative function from training data with the similarity function of canonical correlation analysis (CCA) [6]. In spirit, MDA can be roughly viewed as to conduct manifolds matching in a novel learned discriminating space under the MMD framework, and the learning purpose of MDA shares the similar LDA inspiration with DCC. Compared with DCC, MDA models image set by manifold, which seems more reliable than the linear subspace representation used in DCC, especially when the image set exhibits significantly large appearance variations. Also, DCC needs an iterative learning, which can not provide closed form solution as the proposed MDA.

4. Experiments

The proposed method is evaluated on two tasks: face recognition with image sets and object categorization. Both tasks are handled as the multi-manifolds classification problem. Each known subject is enrolled with a set of images and modeled as a gallery manifold, while a testing subject is modeled as a probe manifold. The task is to classify an unknown probe manifold to one of the gallery manifolds, by following the testing procedure in Sec.2.4.

4.1. Databases

For face recognition task, three datasets are considered: Honda/UCSD [16], CMU MoBo [10], and YouTube Celebrities [14], to ensure extensive evaluations of different methods. For object categorization, we use the benchmark database ETH-80 [17]. We give below a brief description for each of these databases.

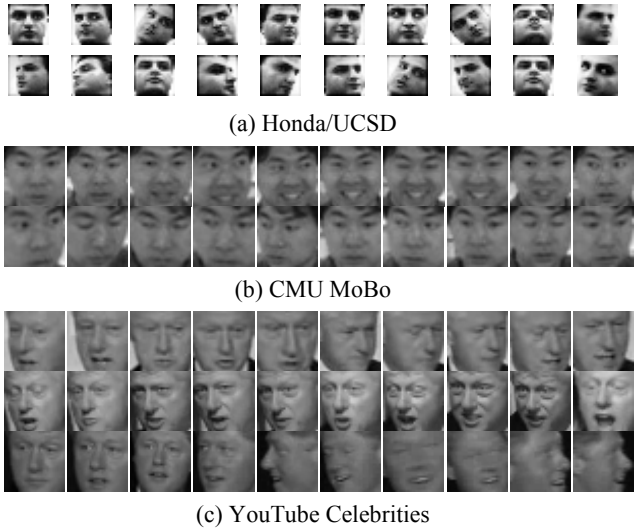
Honda/UCSD: This database is the benchmark for face recognition based on image sets or video [14], [16], [26], [31]. We use its first subset, which consists of 59 video sequences of 20 different persons (each person has at least 2 videos). Each video contains about 400 frames covering large variations in out-of-plane (left/right and up/down) head movement as well as in facial expression. Following the similar settings to that in [16], [26], we use a cascaded face detector [25] to collect faces in each video, and then resize each face to 20×20 gray image. Histogram equalization is employed to eliminate the lighting effects. Some examples are illustrated in Fig. 3(a).

CMU MoBo: The MoBo (Motion of Body) database was originally collected for human identification from distance. There are 96 sequences of 24 different subjects. Each subject has 4 sequences captured in different walking situations: holding a ball, fast walking, slow walking, and walking on the incline. Each sequence has 300 frames. Face images are obtained in the same way as above. The size of the resulted facial images is 30×30 pixels. Fig. 3(b) shows one person's examples from his 2 sequences.

YouTube Celebrities: This dataset was collected by Kim *et al*. [14] for face tracking and recognition in real world applications. The dataset contains 1910 video clips of 47 celebrities, mostly actors/actresses and politicians, from YouTube. Most of the videos are low resolution and recorded at high compression rates, which leads to noisy, low-quality image frames. Each clip contains hundreds of frames. Compared with Honda and MoBo, this database is much more challenging as the videos exhibit very large variations in face pose, illumination, expression, and other conditions. Faces are obtained as above and resized to 30 ×30. Fig. 3(c) shows 3 example clips of one person.

ETH-80: This database contains images of the following eight categories: apples, cars, cows, cups, dogs, horses, pears and tomatoes, as illustrated in Fig. 4(a). Each category includes 10 objects (e.g., 10 different dogs as shown in Fig.4(b)), with 41 images of different views. Object categorization is to classify a set of objects into a group of known categories (e.g., apples, cars, etc.).

On all of four datasets, we conducted ten-fold cross validation experiments, i.e., 10 randomly selected training and testing combinations, for reporting identification rates. In particular, for both Honda and MoBo, each person has one video clip for training and the rest clips for testing. For YouTube, note that each person has, on average, a total of 41 clips. In each of the ten-fold cross validations, one person has 3 randomly chosen clips for training and 6 for testing. This has enabled the whole testing sets in our experiment to cover all of the 1910 clips in the database. For ETH-80, each category has 5 objects for training and the other 5 objects for testing.

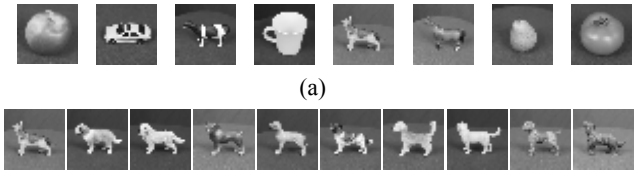


(a) Honda/UCSD

(b) CMU MoBo

(c) YouTube Celebrities

Figure 3: Examples of three face databases. Each row contains representative facial images from one video clip.



(a)

(b)

Figure 4: Example images of the object category database (ETH-80) contain (a) 8 different categories and (b) 10 different objects for one category.

4.2. Experimental setting

We compared the performance of the following two categories of methods, as discussed in Sec.3:

- sample-based methods, include
  1. Nearest neighbor (NN) matching by LDA, i.e., Fisherface [2], as the baseline sample-based method;
  2. Kernel Fisher Discriminant (KFD) [19];
  3. Marginal Fisher Analysis (MFA) [28];
- set-based methods, include
  4. Mutual Subspace Method (MSM) [27], as the baseline set-based method;
  5. Manifold-Manifold Distance (MMD) [26];
  6. Discriminant Canonical Correlations (DCC) [15];
  7. The proposed MDA method.

To compare different algorithms, important parameters of each method were optimized empirically within a wide range, based on the performance in ten-fold cross validation. For the sample-based methods (LDA, KFD and MFA), to avoid singularity problems, we adopted the techniques in accordance with [2], [28]. More specifically, PCA was applied to keep 95% data energy. Gaussian kernel

was used in the experiments of KFD. The best dimension of LDA and KFD subspace was set to classes' number minus 1. The dimension of MFA was chosen around 50 to 70. These three methods all determined the identity of the probe set using majority voting scheme as previous works.

For the set-based methods (MSM, MMD and DCC), we followed the setting similar to that of [15], [26]. In both MSM and DCC, PCA was performed to learn the linear subspace of each image set, and the dimension of the subspace was around 10 by preserving 95% data energy. All canonical correlations were exploited to measure the set similarity. Here, note that, the learning of DCC needs at least two sets for each class to construct the within-class sets. However, as stated in Sec. 4.1, in our experiment the two face databases, Honda and MoBo, had only one training set for each class. Similar to the setting in [15], we just divided each training set into two ones randomly to satisfy the DCC learning requirement. For MMD, we also followed the same parameter settings as specified in [26].

The important parameters in the proposed MDA include: (i) the number of local models, ni, in Eq.(1). As discussed in Sec.2.2, an appropriate value can be chosen by computing the first position on the nonlinearity score curve in Fig.2(b) whose first-order derivative approximates zero. (ii) The number of between-class NN local models, k', in the MDA training step to construct penalty graph (refer to Sec.2.3). (iii) The dimension of MDA embedding space, l, used in the MDA testing procedure. Typical values of these parameters on the four datasets are listed in Table 2.

4.3. Results and analysis

Table 3 summarizes the recognition results of all comparative methods on the four different databases. Each reported rate is an average over the ten runs of cross validation. Overall, MDA achieves superior performances in most aspects of testing. We next highlight some observations about these experiments.

1) Among all the experiments, sample-based methods are generally inferior to set-based methods. This finding is similar to that in [15], [26]. Since they do not exploit any set information, sample-based methods can not be expected to yield superior performance in the set classification tasks.

2) The four set-based methods show distinct performance due to their properties. Both MSM and MMD concern with matching image sets in the original data space. Lack of discriminative learning makes them less appealing than DCC and the proposed MDA. Moreover, MSM is much inferior to MMD mainly because it simply represents the complex image set as a linear subspace. This also explains the superiority of MDA over DCC to some extent.

3) Among four databases, all methods yield the worst recognition rates in YouTube. Considering that all videos come from real world in low quality and cover very large variations, data distribution in this dataset is considerably broad, making this result unsurprising. The proposed MDA outperforms other competing methods with the highest rate of 67.2%. This is compared with 71.24% reported in [14]. However, the method in [14] is video-based, i.e., it exploits video dynamic features. Also, the experimental settings in that paper have many differences from those in this work.

Table 2. Typical values of MDA parameters

Dataset	parameter		
	ni	k'	l
Honda/UCSD	6	3	70
CMU MoBo	8	3	50
YouTube	9	10	70
ETH-80	2	5	10

4) From the results on ETH-80, it is worth noting that the recognition rates of MSM and DCC show similar behavior to those of [15], with DCC delivering the best performance in this database. The proposed MDA is slightly inferior to DCC. This may be due to the characteristics of the database. As stated above, each object set in this database contains only 41 images. The critical sparsity of the image set may have deteriorated the manifold representations in our MDA more or less. Nonetheless, MDA outperformed all the other competing methods.

5. Conclusion

We have proposed a novel discriminative method for classification-oriented multi-manifolds learning. Based on the idea of maximizing "manifold margin", our approach is theoretically and practically appealing. The promising experimental results on face recognition and object categorization further demonstrate that, the proposed MDA is convincingly applicable to image set classification problems, and comparable to the state-of-the-art methods.

For future work, we are now exploring kernel extension of the concept of MDA, which would capture higher-order discriminatory information in image sets. Applications to incremental learning with new training sets are also of our main interests.

Acknowledgements

The authors are extremely grateful to Dr. Shiguang Shan and Prof. Wen Gao for numerous invaluable suggestions during the development of this work and the preparation of the paper. This paper is partially supported by NSFC under contracts Nos.60533030, U0835005, 60803084, 60872077; National Basic Research Program of China (973 Program) under contract 2009CB320902; Hi-Tech Research and Development Program of China under contract No. 2007AA01Z163; Co-building Program of Beijing Municipal Education Commission; and ISVISION Technology Co. Ltd.

Table 3. Evaluations by ten-fold cross validation on four datasets

Dataset	The mean and standard deviation of recognition rates of different methods						
	LDA	KFD	MFA	MSM	MMD	DCC	MDA
Honda/UCSD	0.789 ± 0.01	0.815 ± 0.01	0.838 ± 0.01	0.887 ± 0.04	0.971 ± 0.02	0.980 ± 0.01	1.000 ± 0.00
CMU MoBo	0.885 ± 0.01	0.898 ± 0.03	0.885 ± 0.02	0.848 ± 0.03	0.935 ± 0.02	0.903 ± 0.05	0.965 ± 0.02
YouTube	0.573 ± 0.03	0.607 ± 0.01	0.594 ± 0.02	0.595 ± 0.04	0.640 ± 0.04	0.667 ± 0.04	0.672 ± 0.04
ETH-80	0.673 ± 0.02	0.811 ± 0.02	0.801 ± 0.02	0.833 ± 0.04	0.850 ± 0.07	0.908 ± 0.05	0.890 ± 0.02

References

[1] O. Arandjelović, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face Recognition with Image Sets Using Manifold Density Divergence. *CVPR*, pp. 581–588, 2005.

[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *PAMI*, 19(7):711–720, 1997.

[3] M. Belkin and P. Niyogi. Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. *NIPS*, vol. 14, pp. 585–591, 2001.11.

[4] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. *Journal of Machine Learning Research*, vol. 7, pp. 2399-2434, 2006.

[5] M. Bressan and J. Vitrià. Nonparametric Discriminant Analysis and Nearest Neighbor Classification. *Pattern Recognition Letters*, vol. 24, no. 15, pp. 2743-2749, 2003.

[6] Å. Björck and G. H. Golub. Numerical Methods for Computing Angles between Linear Subspaces. *Mathematics of Computation*, 27(123):579–594, 1973.

[7] H.T. Chen, H.W. Chang, and T.L. Liu. Local Discriminant embedding and its variants. In *Proc. CVPR*, 2005.

[8] F.R.K. Chung. Spectral Graph Theory. *Proc. Regional Conf. Series in Math.*, no. 92, 1997.

[9] W. Fan and D.-Y. Yeung. Locally Linear Models on Face Appearance Manifolds with Application to Dual-Subspace Based Classification. *CVPR*, pp. 1384–1390, 2006.

[10] R. Gross and J. Shi. The CMU Motion of Body (MoBo) database. Technical Report CMU-RI-TR-01-18, Robotics Institute, Carnegie Mellon University, June 2001.

[11] A. Hadid and M. Pietikäinen. From Still Image to Video-Based Face Recognition: An Experimental Analysis. *FG*, pp. 813–818, 2004.

[12] X. He, S. Yan, Y.Hu, P. Niyogi, H. Zhang. Face Recognition Using Laplacianfaces. *PAMI*, 27(3):328–340, 2005.

[13] L. Kaufman, P. J. Rousseeuw. Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, New York. 1990.

[14] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley. Face Tracking and Recognition with Visual Constraints in Real-World Videos. *CVPR* 2008.

[15] T.K. Kim, J. Kittler, and R. Cipolla. Discriminative Learning and Recognition of Image Set Classes Using Canonical Correlations. *PAMI*, vol.29, no.6, pp.1005–1018, 2007.

[16] K.C. Lee, J. Ho, M.H. Yang, and D. Kriegman. Video-Based Face Recognition Using Probabilistic Appearance Manifolds. *CVPR*, pp. 313–320, June 2003.

[17] B. Leibe and B.Schiele. Analyzing Appearance and Contour Based Methods for Object Categorization. *CVPR* 2003.

[18] H. Li, T. Jiang, and K. Zhang. Efficient and Robust Feature Extraction by Maximum Margin Criterion. *IEEE Trans. Neural Networks* 17 (1), 2006, pp. 157–165.

[19] S. Mika, G. Rätsch, and K.-R. Müller. A Mathematical Programming Approach to the Kernel Fisher Algorithm. *NIPS*, vol. 13, pp. 591–597, 2000.

[20] S. Roweis and L. Saul. Nonlinear Dimensionality Reduction by Locally Linear Embedding. *Science*, 290(5500):2323–2326, December 2000.

[21] G. Shakhnarovich, J. W. Fisher, and T. Darrell. Face Recognition from Long-term Observations. *ECCV*, volume 3, pp. 851–865, 2002.

[22] Y. Song, F. Nie, and C. Zhang. Semi-supervised Sub-manifold Discriminant Analysis. *Pattern Recognition Letters*. vol. 29, Issue 13, October 2008, pp. 1806-1813.

[23] J. Stallkamp, H.K. Ekenel, R. Stiefelhagen. Video-based Face Recognition on Real-World Data. *ICCV* 2007.

[24] J. Tenenbaum, V. Silva, and J. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323,December 2000.

[25] P. Viola and M. Jones. Robust Real-Time Face Detection. *Int'l J. Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[26] R. Wang, S. Shan, X. Chen, W. Gao. Manifold-Manifold Distance with Application to Face Recognition based on Image Set. *CVPR* 2008.

[27] O. Yamaguchi, K. Fukui, K. Maeda. Face Recognition Using Temporal Image Sequence. *FG*, pp. 318–323, 1998.

[28] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin. Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. *PAMI*, 29(1): 40–51, 2007.

[29] S. Yan, H. Zhang, Y. Hu, B. Zhang, Q. Cheng. Discriminant Analysis on Embedded Manifold. *Proc. Eighth European Conf. Computer Vision*, vol. 1, pp. 121-132, May 2004.

[30] J. Yang, D. Zhang, J.-Y. Yang, and B. Niu. Globally maximizing, locally minimizing: Unsupervised discriminant projection with applications to face and palm biometrics. *PAMI*, 29(4): 650–664, 2007.

[31] Y. Zhao, S. Xu, Y. Jia. Discriminant Clustering Embedding for Face Recognition with Image Sets. *ACCV* 2007.

[32] S. Zhou and R. Chellappa. Probabilistic Human Recognition from Video. *ECCV*, volume 3, pages 681–697, 2002.

[33] M. Zhu and A.M. Martinez. Subclass Discriminant Analysis. *PAMI*, 28(8): 1274-1286, Aug. 2006.