Manifold–Manifold Distance and Its Application to Face Recognition With Image Sets

Ruiping Wang, *Member, IEEE*, Shiguang Shan, *Member, IEEE*, Xilin Chen, *Senior Member, IEEE*, Qionghai Dai, *Senior Member, IEEE*, and Wen Gao, *Fellow, IEEE*

***Abstract*— In this paper, we address the problem of classifying image sets for face recognition, where each set contains images belonging to the same subject and typically covering large variations. By modeling each image set as a manifold, we formulate the problem as the computation of the distance between two manifolds, called manifold–manifold distance (MMD). Since an image set can come in three pattern levels, point, subspace, and manifold, we systematically study the distance among the three levels and formulate them in a general multilevel MMD framework. Specifically, we express a manifold by a collection of local linear models, each depicted by a subspace. MMD is then converted to integrate the distances between pairs of subspaces from one of the involved manifolds. We theoretically and experimentally study several configurations of the ingredients of MMD. The proposed method is applied to the task of face recognition with image sets, where identification is achieved by seeking the minimum MMD from the probe to the gallery of image sets. Our experiments demonstrate that, as a general set similarity measure, MMD consistently outperforms other competing nondiscriminative methods and is also promisingly comparable to the state-of-the-art discriminative methods.**

***Index Terms*— Face recognition with image sets, hierarchical divisive clustering, manifold–manifold distance (MMD), principal angles, set similarity measure.**

I. Introduction

In TRADITIONAL face recognition task, subjects of interest are trained and recognized from only a few samples. Recently, with the increase of available video cameras and large capacity storage media, many new applications such as visual surveillance and video retrieval, in which the image quantity of each subject of interest for both training and testing can be very large, are emerging. For example, as shown in Fig. 1, a great number of images for each known subject have been able to be collected from video sequences or photo album, and recognition can also be conducted with a set of probe images rather than a single probe image. In other words, recognition can be formulated as matching a probe image set against all the gallery image sets each representing one subject. We call this category of face recognition task as Face Recognition with Image Sets (FRIS) problem. In the FRIS task, the images in each set are generally of large amount and cover large variations of the subject, due to subject's pose changes, non-rigid deformations or different lighting conditions. By efficiently exploiting set information, more robust face recognition can be expected under more realistic conditions [1]–[3].

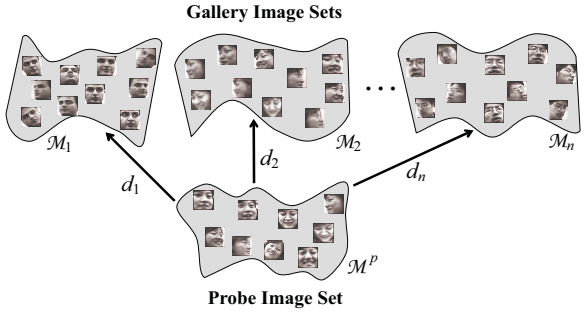

Fig. 1. FRIS, where each subject is enrolled with a gallery image set, and the unknown subject is represented as a probe image set.

Over the past decade, FRIS problem attracted increasing interest in computer vision community [4]–[13]. It is worth pointing out that video-based face recognition [7]–[9], [13] is only a special case of FRIS. In the general scenario of FRIS, the images in the gallery or probe sets are collected not necessarily from consecutive video sequences but possibly from multiple unordered observations of a subject.

A. Previous Work

From the view of image set modeling, related approaches to set matching based face recognition broadly fall into two classes: model-dependent parametric methods and model-independent nonparametric methods. Typical parametric methods include probabilistic modeling method [11] and manifold density divergence [1]. They tend to represent each image set by a parametric distribution function and then measure the

Manuscript received October 10, 2011; revised April 16, 2012; accepted May 24, 2012. Date of publication June 26, 2012; date of current version September 13, 2012. This work was supported in part by the National Basic Research Program of China 973 Program under Contract 2009CB320902; the Natural Science Foundation of China under Contract 61173065, Contract 60833013, and Contract 61025010; and the Beijing Natural Science Foundation, New Technologies and Methods in Intelligent Video Surveillance for Public Security under Contract 4111003. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mary Comer.

R. Wang was with the Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS), Institute of Computing Technology, CAS, Beijing 100190, China. He is now with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: rpwang@mail.tsinghua.edu.cn).

S. Shan and X. Chen are with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: sgshan@ict.ac.cn; xlchen@ict.ac.cn).

Q. Dai is with the Department of Automation, Tsinghua University, Beijing 100084, China (e-mail: qionghaidai@mail.tsinghua.edu.cn).

W. Gao is with the School of Electrical Engineering and Computer Science, Key Laboratory of Machine Perception, Peking University, Beijing 100871, China (e-mail: wgao@pku.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2012.2206039

similarity between two distributions in terms of the Kullback-Leibler Divergence (KLD). In [11], face pattern variations are modeled by relatively simplistic single Gaussian distribution in the face space. For more realistic and satisfactory modeling, Gaussian mixture models (GMM) is used in [1] instead. While parametric methods have shown promising results in lots of applications, they typically need to solve the difficult parameter estimation problem and may have large performance fluctuation in case that the training and the novel test data sets have weak statistical correlations [2], [3].

In comparison, nonparametric methods typically relax the assumption on distribution of the set data, and try to model the image set in more flexible manners. In the earlier work [10], elements in the set, i.e., image samples are treated separately and set matching is conducted by finding the closest pair of samples from the image sets. However, such single sample matching-based methods pay less attention to the global natural data variations across the whole set, making them more sensitive to the effect of outliers. Also, their computational cost is considerably high since they need to compute all pairwise sample distances whenever to compare two image sets.

More recently, a favorable trend is using subspace learning techniques to model the set data variability globally, following the pioneer work [14]. These methods attempt to represent the image set either by linear subspace [3], [5], [12], [15], [16] or by nonlinear manifold [2], [4], [17] and then conduct set classification by comparing subspaces or manifolds based on different similarity measures.

Generally, nonparametric methods can be categorized roughly into two groups: one group focuses on how to define the set similarity measure as in [2], [14], [16], [17], while the other group pays more attention to learning discriminative classifier for a given similarity function as in [3], [4], [5], [12], [15]. Among the former group, in [17], representative samples called "exemplars" are extracted from image sets as local models. Set matching is then conducted by measuring the similarity of these exemplars. In [2], [14], principal angles [18], [19] are exploited as the similarity measure of two linear subspaces. To capture the data nonlinearity, [16] proposes a kernel version of principal angles. In the latter group, with subspace modeling of the image sets, [3] and [5] similarly learn a linear discriminant subspace that aims to maximize the class separation in terms of principal angles. To further address complex data distribution, [4] seeks to extract local discriminating information from nonlinear manifolds with local linear model representations. Beyond but closely related to principal angles, the so-called Grassmannian distances [20] are studied in [15] as subspace distances and further utilized for discriminant learning. For similar purpose of handling data nonlinearity, [12] explores a kernel extension of [15] by computing the Grassmannian distances in high dimensional feature space via kernel trick.

In summary, the two classes of approaches, model-dependent parametric and model-independent nonparametric methods, have their own advantages and are applicable to different cases. While the former try to estimate the underlying density distributions confined to low-dimensional subspaces of the data, the latter typically aim to learn the subspaces directly. If the training and testing data share similar statistical properties, the parametric methods are expected to produce better results [1]. For more general set classification tasks, patterns in the training and testing stages might vary significantly due to different collection conditions. In this case, the nonparametric subspace learning methods are more suitable since they impose uniform prior over the space of possible data variations [2], [3]. In practical applications, the choice between the two classes of approaches should be determined by a particular task.

On the nonparametric side, it has been shown that the set of face images acquired under (only) varying illumination conditions forms an intrinsically low-dimensional linear subspace [21]. Moreover, as analyzed in [2], [4], [17], while other data variations (e.g., pose and expression) are involved, face images in the set will exhibit significant nonlinearities. Therefore, it is important to use image set representation flexible enough to deal with the nonlinearity. Assuming the images in each set reside on a nonlinear manifold, the FRIS task can be converted to the problem of matching different manifolds based on some similarity measurements. To our knowledge, the topic of a general similarity or distance measure over manifolds has not been given sufficient attention in the literature before. It is this point that motivates the work in this paper.

B. Our Approach

In this paper, we propose a distance criterion called Manifold to Manifold Distance (MMD) for face recognition with image sets. Based on the local linearity property of manifold, we represent a manifold as a collection of local linear models, each depicted by a subspace. MMD is then converted to integrating the local distances between each pair of subspaces, which respectively comes from one of the two involved manifolds. Preliminary results of the method have been published in [22]. Compared with the conference version, this paper has made three major extensions. First, the method to construct local linear models from manifold is improved in a more effective and flexible way. Second, we provide a more detailed comparison and discussion regarding different possibilities for the definition of MMD. Third, more extensive experiments are carried out to evaluate the method and compare with other state-of-the-art algorithms.

The rest of the paper is organized as follows: we give an overview of the distances among point, subspace and manifold in Section II. In Section III, we highlight the three key ingredients of MMD and propose several possible definitions of MMD. In Section IV, we make comparisons between the proposed MMD and other related works, and give its complexity analysis. Comprehensive experiments are presented in Section V. Finally, we draw conclusions in Section VI.

II. Formulation of Distance Categories

In practical FRIS tasks, the size of an image set can vary from a large number to a single one. To accommodate such different cases, the image set can thus be represented in three possible pattern levels: point (i.e., individual sample), subspace (i.e., linear model spanned by a few samples), and manifold (i.e., nonlinear low-dimensional embedding spanned

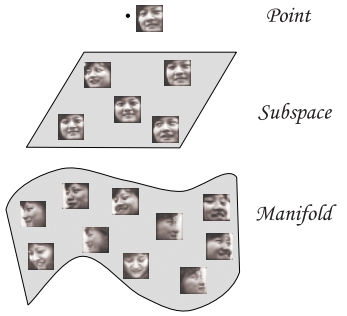

Fig. 2. Hierarchical structure formed by three pattern levels (i.e., point, subspace, and manifold) in face recognition.

by a large number of samples). See Fig. 2 for an illustration. In some sense, the core of pattern classification is the distance computation among these representations. The distances over point and subspace have been well studied in the literature; whereas very few studies have been done on the distance related to manifold.

A. Distances Over Point and Subspace

Hereinafter we always denote points by $\boldsymbol{x}_i, \boldsymbol{y}_i$, subspaces by S_i, and manifolds by $\mathcal{M}_i$. The distances over point and subspace include the following three ones:

1) Point to Point Distance (PPD): denote by $d(\boldsymbol{x}_1, \boldsymbol{x}_2)$ the distance from point $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$. The most commonly used PPD is the Euclidean distance as follows:

$$d(\boldsymbol{x}_1, \boldsymbol{x}_2) = \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|. \tag{1}$$

2) Point to Subspace Distance (PSD): denote by $d(\boldsymbol{x}, S)$ the distance from point $\boldsymbol{x}$ to subspace S. It is generally defined as the so-called L2-Hausdorff distance:

$$d(\boldsymbol{x}, S) = \min_{\boldsymbol{y} \in S} \|\boldsymbol{x} - \boldsymbol{y}\| = \|\boldsymbol{x} - \boldsymbol{x}'\|. \tag{2}$$

In fact, $\boldsymbol{x}'$ is the projection of $\boldsymbol{x}$ in the subspace S, also the nearest point to $\boldsymbol{x}$ in S. Thus, the PSD is actually the PPD from $\boldsymbol{x}$ to its projection $\boldsymbol{x}'$ in S. It is also known as "distance-from-feature-space" (DFFS) in [23].

3) Subspace to Subspace Distance (SSD): denote by $d(S_1, S_2)$ the distance between two subspaces S_1 and S_2. While there is not a unified definition yet to measure the SSD, the concept of principal angles [18], [19] is perhaps the most commonly exploited one due to its favorable performance. Recently, another SSD is proposed in [24], which can be regarded as utilizing the sum of DFFS between the bases of two subspaces.

As known in linear algebra, the single point $\boldsymbol{x}_i$ spans a special linear subspace, i.e., the trivial zero subspace $L\{\boldsymbol{0}\}$, which is centered on $\boldsymbol{x}_i$ and of zero dimensional. In this sense, both PPD and PSD are special cases of SSD.

B. Distances Over Manifold

Our main motivation arises from the fact that local linearity holds everywhere on a globally nonlinear manifold. Thus, a manifold can be modeled by a collection of local linear models, each depicted by a subspace [25]. In general, manifold can be viewed as extending subspace to account for more general and complex data variations. The distances associated with manifold are then related to those defined on subspace. Formally, we denote the i-th component subspace of a manifold $\mathcal{M}$ by C_i, and express $\mathcal{M}$ as a set containing all the C_i:

$$\mathcal{M} = \{C_i : i = 1, 2, \ldots, m\} = \{C_1, C_2, \ldots, C_m\}. \tag{3}$$

where m is the number of local linear subspaces.

1) Point to Manifold Distance (PMD): denote by $d(\boldsymbol{x}, \mathcal{M})$ the distance from point $\boldsymbol{x}$ to manifold $\mathcal{M}$. Similar to PSD, one can define this distance by finding the closest point to $\boldsymbol{x}$ in $\mathcal{M}$ as follows:

$$d(\boldsymbol{x}, \mathcal{M}) = \min_{C_i \in \mathcal{M}} d(\boldsymbol{x}, C_i) = \min_{C_i \in \mathcal{M}} \min_{\boldsymbol{y} \in C_i} \|\boldsymbol{x} - \boldsymbol{y}\| = \|\boldsymbol{x} - \boldsymbol{x}''\|. \tag{4}$$

In analogy to $\boldsymbol{x}'$ in the PSD, here we call $\boldsymbol{x}''$ the projection of $\boldsymbol{x}$ in the manifold $\mathcal{M}$.

2) Subspace to Manifold Distance (SMD): denote by $d(S, \mathcal{M})$ the distance from subspace S to manifold $\mathcal{M}$. It can be defined by seeking the closest subspace to S in manifold $\mathcal{M}$:

$$d(S, \mathcal{M}) = \min_{C_i \in \mathcal{M}} d(S, C_i). \tag{5}$$

It comes that SMD is reduced to SSD in a simple manner similar to that from PSD to PPD.

3) Manifold to Manifold Distance (MMD): denote by $d(\mathcal{M}_1, \mathcal{M}_2)$ the distance between two manifolds $\mathcal{M}_1$ and $\mathcal{M}_2$. With the local linear model representation in (3), MMD can be converted to integrating the distances between pair of subspaces respectively from one of the involved manifolds. See Fig. 3 for a conceptual illustration.

Formally, given two manifolds $\mathcal{M}_1 = \{C_i : i = 1, 2, \ldots, m\}$, $\mathcal{M}_2 = \{C'_j : j = 1, 2, \ldots, n\}$, we formulate MMD as follows:

$$\begin{aligned} d(\mathcal{M}_1, \mathcal{M}_2) &= \sum\nolimits_{i=1}^{m} \sum\nolimits_{j=1}^{n} f_{ij} d\left(C_i, C'_j\right), \\ \text{s.t.} \quad & \sum\nolimits_{i=1}^{m} \sum\nolimits_{j=1}^{n} f_{ij} = 1, f_{ij} \geq 0. \end{aligned} \tag{6}$$

In this general formulation, MMD comes in the form of a weighted average of pairwise SSDs, i.e., $d(C_i, C'_j)$.

It has been figured out that point is a special case of subspace. Similarly, subspace can be viewed as a special case of manifold under the formulation in (3). Therefore, the three pattern levels form a hierarchical structure and all the six distances can be formulated in a general multi-level MMD framework.

III. Manifold–Manifold Distance

From Fig. 3 and (6), one can find that there are three key ingredients in MMD: (i) *local linear model construction*, i.e., the component subspaces C_i, C'_j, (ii) *local model distance measure*, i.e., the SSD $d(C_i, C'_j)$, and (iii) *global integration of local distances*, i.e., the choice of the weights f_{ij}. In this section, we present details of these ingredients and extensive investigations on their various configurations.

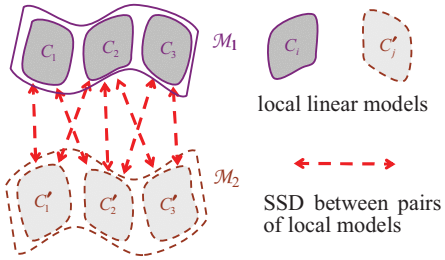

Fig. 3. Conceptual illustration of MMD. The distance between two manifolds, $\mathcal{M}_1$ and $\mathcal{M}_2$, is converted to integrate the distances between their corresponding local linear models, C_i and C'_j.

A. Local Linear Model Construction

To extract local models from manifold, previous works generally use classical clustering methods, e.g., K-means [2], [8], [17] or hierarchical agglomerative clustering (HAC) [4]. They have two main limitations: first, the number of target clusters (i.e. local models) needs to be specified a priori; second, the linearity property of the extracted local models is not guaranteed explicitly. To overcome these limitations, we propose an effective and efficient clustering algorithm for adaptively constructing multi-level local models with explicit linearity guarantee.

We first introduce the concept of Maximal Linear Patch (MLP). In brief, an MLP on the manifold is defined as a local linear patch, whose nonlinearity degree is elegantly measured by the deviation between Euclidean distance and geodesic distance. To construct MLPs, our conference paper [22] has derived a one-shot sequential clustering method that can only yield MLPs for a pre-specified nonlinearity degree and might suffer from the problem of unbalanced clusters. In this paper, we further combine the merits of MLP and hierarchical clustering method. Since in most cases the appropriate number of clusters is much smaller than the number of data samples, here we explore the more efficient top-down hierarchical divisive clustering (HDC) rather than the bottom-up HAC manner used in previous work [4]. As the quality of the cluster (i.e., MLP) is measured in terms of nonlinearity degree, we call the clustering method Linearity-constrained HDC (L-HDC). Our recent works [26], [27] have conducted some preliminary study on the method. This paper will give more extensive investigation and experimental validation.

Formally, given a manifold $\mathcal{M}$ with its data set $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$, where $\boldsymbol{x}_i \in \mathrm{R}^D$ is a D-dimensional column vector, and N is the sample number. We aim to extract a collection of MLPs $\boldsymbol{X}^{(i)}$ from $\boldsymbol{X}$, i.e.,

$$\boldsymbol{X} = \bigcup_{i=1}^{m} \boldsymbol{X}^{(i)};$$
$$\boldsymbol{X}^{(i)}|_{i=1}^{m} = \left\{\boldsymbol{x}_1^{(i)}, \boldsymbol{x}_2^{(i)}, \ldots, \boldsymbol{x}_{N_i}^{(i)}\right\}, \quad \left(\sum_{i=1}^{m} N_i = N\right). \tag{7}$$

Each MLP $\boldsymbol{X}^{(i)}$ is then modeled as a linear subspace C_i to obtain the local linear model representation in (3).

Firstly, the pairwise Euclidean distance matrix $\boldsymbol{D}_E$ and geodesic distance matrix $\boldsymbol{D}_G$, based on k-NN graph, are computed as in [28]. Then a matrix holding distance ratios is obtained as: $\boldsymbol{R}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{D}_G(\boldsymbol{x}_i, \boldsymbol{x}_j)/\boldsymbol{D}_E(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Clearly, these three matrices are all of size $N \times N$. Since geodesic distance is always not smaller than Euclidean distance, $\boldsymbol{R}(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 1$ holds for any entry of $\boldsymbol{R}$. Besides, another matrix $\boldsymbol{H}$ of size $k \times N$ is also constructed, each column $\boldsymbol{H}(:, j)(j = 1, \ldots, N)$ holding the indices of k nearest neighbors of the data point $\boldsymbol{x}_j$. Note that, as a byproduct of the computation of $\boldsymbol{D}_E$ and $\boldsymbol{D}_G$, the construction of $\boldsymbol{H}$ requires no extra computation. To measure the nonlinearity degree of an MLP, $\boldsymbol{X}^{(i)}$ $(i = 1, 2, \ldots, m)$, we define the following *nonlinearity score function*:

$$\beta^{(i)} = \frac{1}{N_i^2} \sum_{p=1}^{N_i} \sum_{q=1}^{N_i} \boldsymbol{R}\left(\boldsymbol{x}_p^{(i)}, \boldsymbol{x}_q^{(i)}\right). \tag{8}$$

Algorithm 1 Linearity-Constrained HDC (L-HDC)

1 Initialization: $\boldsymbol{X}^{(1)} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$, $m = 1$. Compute the nonlinearity score $\beta^{(1)}$ according to (8).
2 Choose $\boldsymbol{X}^{(i)}$ $(i \in \{1, 2, \ldots, m\})$ with the largest score $\beta^{(i)}$ as the *parent* cluster. Split $\boldsymbol{X}^{(i)}$ as follows:
2.1 According to geodesic distance $\mathbf{D}_G$, select two furthest seed points, $\boldsymbol{x}_l$ and $\boldsymbol{x}_r$ from $\boldsymbol{X}^{(i)}$.
2.2 Initialize two *child* clusters: $\boldsymbol{X}_l^{(i)} = \{\boldsymbol{x}_l\}$, $\boldsymbol{X}_r^{(i)} = \{\boldsymbol{x}_r\}$. Update: $\boldsymbol{X}^{(i)} \leftarrow \boldsymbol{X}^{(i)} \backslash \{\boldsymbol{x}_l, \boldsymbol{x}_r\}$.
2.3 while $(\boldsymbol{X}^{(i)} \neq \emptyset)$ do
2.3.1 For current $\boldsymbol{X}_l^{(i)}$, construct its neighbor points set, denote by P_l. According to $\boldsymbol{H}$, P_l gathers the k-NN samples of all the points in $\boldsymbol{X}_l^{(i)}$.
2.3.2 For current $\boldsymbol{X}_r^{(i)}$, construct its neighbor points set P_r in the similar way to step 2.3.1.
2.3.3 Sequentially update:
$\boldsymbol{X}_l^{(i)} \leftarrow \boldsymbol{X}_l^{(i)} \cup (P_l \cap \boldsymbol{X}^{(i)})$, $\boldsymbol{X}^{(i)} \leftarrow \boldsymbol{X}^{(i)} \backslash (P_l \cap \boldsymbol{X}^{(i)})$;
$\boldsymbol{X}_r^{(i)} \leftarrow \boldsymbol{X}_r^{(i)} \cup (P_r \cap \boldsymbol{X}^{(i)})$, $\boldsymbol{X}^{(i)} \leftarrow \boldsymbol{X}^{(i)} \backslash (P_r \cap \boldsymbol{X}^{(i)})$.
2.4 $\boldsymbol{X}^{(i)}$ is split into two smaller ones: $\boldsymbol{X}_l^{(i)}$ and $\boldsymbol{X}_r^{(i)}$. Update: $m \leftarrow m + 1$, compute $\beta_l^{(i)}$ and $\beta_r^{(i)}$.
3 The splitting procedure continues until the nonlinearity score $\beta^{(i)}$ in step 2 is less than a *threshold* δ.

The L-HDC is formulated as Algorithm 1. Its basic procedure is that, in the first level, all samples are initiated as a singleton MLP (cluster). Then, in each new level, the MLP in the parent level with the largest nonlinearity degree will be split into two smaller ones with decreased degrees. Finally, we are able to obtain multi-level MLPs with different nonlinearity degrees. Note that the threshold δ in step 3 controls the termination of the algorithm and thus the number of final clusters, i.e., m, as well as their nonlinearity degrees. A larger δ implies fewer clusters but larger linearity deviation, and vice versa. Obviously, the complete clustering hierarchy can be produced whenever δ is specified to any value less than 1, since all $\beta^{(i)}$'s are larger than 1. It can be observed that most steps of Algo.1 are access operations against existing matrices computed in advance. Although it involves some iterative steps, the algorithm runs very efficiently nevertheless. We will give detailed complexity analysis of the algorithm in Section IV-B.

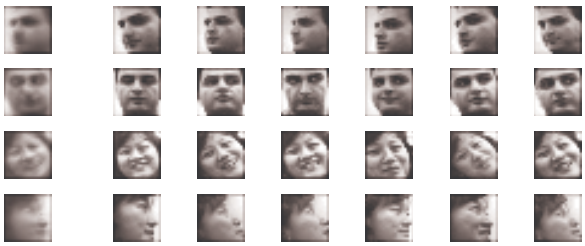

Fig. 4. Local models of the face data. Each row shows a local model with the sample mean (first column), that is, exemplar, and six representative samples. The first and second rows belong to one individual, and the third and fourth rows belong to another individual.

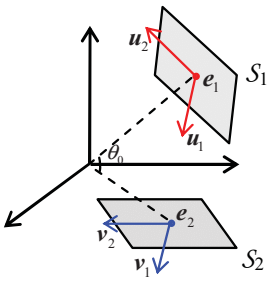

Fig. 5. Conceptual illustration of SSD based on fusing the variation and exemplar distance measures. S_1 and S_2 are two subspaces, which are 2-D planes here. The variation distance is measured by principal angles between the principal axes $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$, which are aligned to obtain the canonical vectors $\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_2]$ and $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2]$, respectively, through an orthogonal transformation (see text for details). The exemplar distance is measured by the angle θ_0 between two exemplars $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$.

Fig. 4 gives some clusters constructed for two of the individuals in Fig. 1. We can observe that samples in a single cluster exhibit only slight (mostly linear) variations in appearance from their mean. Hereinafter, we call the sample mean of each cluster "exemplar", which can represent the samples in the cluster to some extent.

The extracted clusters (i.e., $\boldsymbol{X}^{(i)}$'s) are then represented by linear subspaces to obtain the final local models. Principal component analysis (PCA) is employed for its simplicity and efficiency. For each local model C_i, we denote its sample mean (i.e., exemplar) by $\boldsymbol{e}_i$ and corresponding principal component matrix by $\boldsymbol{P}_i \in \mathrm{R}^{D\times d_i}$ that is computed as the leading eigenvectors of the covariance matrix and forms a set of orthonormal basis of the subspace. Here d_i denotes the PCA subspace dimension. Since the subspace (or local model) is spanned by a set of samples, $\boldsymbol{e}_i$ and $\boldsymbol{P}_i$ play different roles to jointly describe the local model: the former characterizes the data sample itself, and the latter characterizes the data variation modes.

B. Local Model Distance Measure

With the local models constructed above, we can use SSD to measure their distance. Intuitively, a reasonable and complete SSD should take into account both the principal axes $\boldsymbol{P}_i$ and the sample mean $\boldsymbol{e}_i$. As shown in Fig. 5, $\boldsymbol{e}_i$ tells the position of the subspace located in the global observation data space, and $\boldsymbol{P}_i$ tells the spanning directions of the subspace. However, the most commonly exploited SSD, i.e. principal angles, is only associated with $\boldsymbol{P}_i$ and thus merely reflects the difference in the variation modes. We call this distance as "variation based measure". On the contrary, several methods [4], [17] used only $\boldsymbol{e}_i$ to compute the local model distance. We call their distance as "exemplar based measure". Obviously, the two types of distance measures respectively emphasize one side of the coin. To incorporate the mean information when exploiting principal angles for face recognition, [5], [14] have made a compromise by performing PCA with the correlation matrix rather than the covariance matrix. Nonetheless, this scheme makes their resulting subspace basis a mixed version of mean and variance, which does not have as pure and clear physical meaning as ours. In the following, after a brief review of principal angles and previous SSD definitions, we give a new formulation of SSD, which is directly derived from principal angles, and fuses seamlessly both the variation and exemplar based measures.

1) Principal Angles: For two subspaces S_1 and S_2, denote their corresponding exemplars by $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, and orthonormal bases by $\boldsymbol{P}_1 \in \mathrm{R}^{D\times d_1}$, $\boldsymbol{P}_2 \in \mathrm{R}^{D\times d_2}$, where d_1 and d_2 are the subspace dimensions. Principal angles $0 \le \theta_1 \le \theta_2 \le \cdots \le \theta_r \le (\pi/2)$ between the two subspaces S_1 and S_2 are uniquely defined as the minimal angles between any two vectors of the subspaces [19]:

$$\cos\theta_k = \max_{\boldsymbol{u}_k\in S_1}\max_{\boldsymbol{v}_k\in S_2} \boldsymbol{u}_k^T\boldsymbol{v}_k$$
$$\text{s.t. } \boldsymbol{u}_k^T\boldsymbol{u}_k = \boldsymbol{v}_k^T\boldsymbol{v}_k = 1; \quad \boldsymbol{u}_k^T\boldsymbol{u}_i = \boldsymbol{v}_k^T\boldsymbol{v}_i = 0 \quad (i = 1, 2, \ldots, k-1). \tag{9}$$

where $r = \min(d_1, d_2)$. Here in (9), $\boldsymbol{u}_k$ and $\boldsymbol{v}_k$ are called the k-th pair of canonical vectors, the first constraint requires these vectors to be normalized, and the second requires the canonical vectors in each subspace to be orthogonal. Intuitively, the first pair of canonical vectors corresponds to the most similar modes of variation of two linear subspaces; every next pair to the most similar modes orthogonal to all previous ones. The smaller the maximum principal angle is, the closer the two subspaces are.

To calculate the principal angles, a numerically stable algorithm proposed in [18] is based on Singular Value Decomposition (SVD). In the method, the SVD of $\boldsymbol{P}_1^T\boldsymbol{P}_2$ is first computed as follows:

$$\boldsymbol{P}_1^T\boldsymbol{P}_2 = \boldsymbol{Q}_1\boldsymbol{\Lambda}\boldsymbol{Q}_2^T \tag{10}$$

where $\boldsymbol{\Lambda} = diag(\sigma_1, \ldots, \sigma_r)$, $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ are two orthogonal matrices. The singular values $\sigma_1, \ldots, \sigma_r$ are just the cosines of the principal angles, i.e. the so-called canonical correlations:

$$\cos\theta_k = \sigma_k, \qquad k = 1, 2, \ldots, r. \tag{11}$$

The associated canonical vectors are $\boldsymbol{U} = \boldsymbol{P}_1\boldsymbol{Q}_1 = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{d_1}]$ and $\boldsymbol{V} = \boldsymbol{P}_2\boldsymbol{Q}_2 = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{d_2}]$, which are obtained by aligning the two principal axes $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ respectively through an orthogonal transformation, as shown in Fig. 5. One may find that a previous work [29] has also employed PCA based local models and then globally aligned the local PCA subspaces. However, their alignment is for single manifold dimensionality reduction, while ours is for comparing a pair of local models from two manifolds.

2) Previous Work on SSD: Based on principal angles, various subspace distances have been defined in the literature. For example, in the pioneering study named Mutual Subspace Method (MSM) [14], only the smallest principal angle θ_1 is used to define a distance called *Max Correlation* as follows:

$$d_{Max}(S_1, S_2) = (1 - \cos^2\theta_1)^{1/2} = \sin\theta_1. \quad (12)$$

In contrast, another distance called *Min Correlation* is similarly defined using the largest principal angle θ_r:

$$d_{Min}(S_1, S_2) = (1 - \cos^2\theta_r)^{1/2} = \sin\theta_r. \quad (13)$$

Both above distances depend highly on the probability distribution of the principal angles and are effective only in some specific cases respectively, as noted in [15]. Consequently, another distance called *Projection metric*, which uses all the principal angles, was proposed as follows [20]:

$$d_P(S_1, S_2) = \left(\sum_{k=1}^{r} \sin^2\theta_k\right)^{1/2} = \left(r - \sum_{k=1}^{r} \cos^2\theta_k\right)^{1/2}. \quad (14)$$

As advocated in [15], projection metric satisfies the metric axioms and shows intermediate characteristics between the above two distances. For other possible definitions of SSD, please refer to [15], [20], [24] for detailed overview.

3) Proposed SSD Definition: The projection metric in (14) provides a reasonable *variation distance measure* between the two subspaces S_1 and S_2. To derive a better SSD that takes both sample means and variation modes into account, we need to define an exemplar based measure and combine it with projection metric. One possible choice might be the classical PPD in (1). However, there seems no direct way to combine the Euclidean distance in (1) with the projection metric in (14) since they come in different forms. Hence, we resort to the correlation measure that has been widely exploited in face recognition [30]. As shown in Fig. 5, the correlation of the two exemplars $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$, is the cosine of their angle θ_0. We then define an *exemplar distance measure*:

$$d_E(S_1, S_2) = (1 - \cos^2\theta_0)^{1/2} = \sin\theta_0,$$
$$\text{where } \cos\theta_0 = \boldsymbol{e}_1^T\boldsymbol{e}_2/||\boldsymbol{e}_1|| \cdot ||\boldsymbol{e}_2||. \quad (15)$$

With the two distance measures in (14) and (15), we can fuse them in a seamless manner and reach the following *formal definition of SSD*:

$$d(S_1, S_2) = \left(\sin^2\theta_0 + \frac{1}{r}\sum_{k=1}^{r} \sin^2\theta_k\right)^{1/2}$$
$$= \left(2 - \cos^2\theta_0 - \frac{1}{r}\sum_{k=1}^{r} \cos^2\theta_k\right)^{1/2}. \quad (16)$$

Because different subspace pairs do not necessarily have the same number of principal angles, the factor $1/r$ in (16) mainly serves as a normalized weight to balance the two measures.

When applying to comparing two image sets, the two measures complement each other. Take the three manifolds $\mathcal{M}_1$, $\mathcal{M}_2$ and $\mathcal{M}^p$ in Fig. 1 for example. After constructing local models for each manifold, we compute the distances over these local models. In Fig. 6(a) and (b), we show respectively the closest local model pair from $\mathcal{M}_1$ and $\mathcal{M}^p$ (from different individuals), and that from $\mathcal{M}_2$ and $\mathcal{M}^p$ (from the same individual).

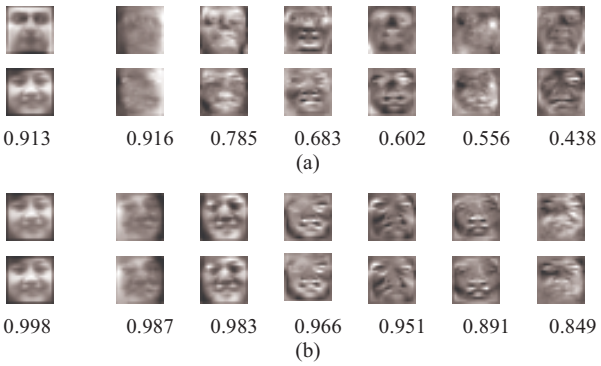

Fig. 6. Local model similarity. (a) Local model pair from different individuals. (b) Local model pair from the same individual. In both (a) and (b), the first column shows the exemplars of the two local models, and other columns show the first six canonical vectors in turn. From their correlation similarity shown below the images, we can see that every pair of canonical vectors captures similar variation modes well. Furthermore, the similarities of the same individual are clearly larger than those of different persons, which shows expected discrimination.

The fusion of exemplar and variation distances in (16) differs from our initial formulation in the conference version [22], where a simple weighted average combination was derived from a somewhat ad hoc intuition. In this study, we have found the improved definition in (16) is not only theoretically more appealing but also experimentally more effective.

C. Global Integration of Local Distances

Now we come to the last component of MMD, i.e., to choose the weights f_{ij} in (6). While it seems a many-to-many matching problem, our FRIS scenario has its special properties. To match the two sets as the same class, the most effective solution would be to find the common views and measure their similarity [3], [10], i.e., rather than matching each pair of local models from two manifolds, those neighboring pairs deserve more emphasis.

Following the notation in Section II-B, given two manifolds $\mathcal{M}_1 = \{C_i : i = 1, 2, \ldots, m\}$, $\mathcal{M}_2 = \{C'_j : j = 1, 2, \ldots, n\}$, we first define two indicator functions as follows:

$$N(i) = \arg\min_j d(C_i, C'_j), \qquad j = 1, 2, \ldots, n,$$
$$N'(j) = \arg\min_i d(C_i, C'_j), \qquad i = 1, 2, \ldots, m. \quad (17)$$

Here, $N(i)$ defined for $\mathcal{M}_1$ indicates the NN's (nearest neighbor) index of the local model C_i in $\mathcal{M}_2$. Similarly, $N'(j)$ defined for $\mathcal{M}_2$ indicates the NN's index of the local model C'_j in $\mathcal{M}_1$. Then, we can obtain a set A containing all the NN local model pairs:

$$A = A_1 \cup A_2, \text{ where}$$
$$A_1 = \left\{(C_i, C'_{N(i)})|_{i=1}^m\right\}, \quad A_2 = \left\{(C_{N'(j)}, C'_j)|_{j=1}^n\right\}. \quad (18)$$

Note that, although the NN relationship is asymmetric, A_1 and A_2 may still contain some common elements, i.e., $A_1 \cap A_2 \neq$

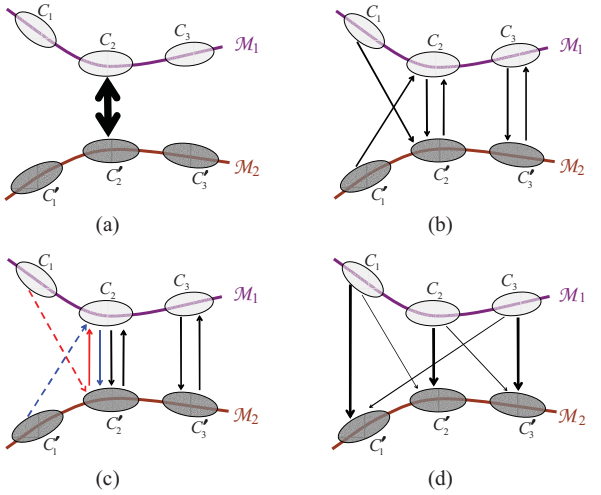

Fig. 7. Different global integration options. Both manifolds have three local models. (a)–(c) Arrows going from each local model to its NN. For example, the NN of C_1 is C'_2, while the NN of C'_2 is C_2 rather than C_1. (d) Arrows going from EMD suppliers to consumers. In all figures, the widths of solid arrows indicate corresponding weights, while dashed arrows in (c) have 0 weights. For example, weight 1 is placed on the only arrow in (a), 1/6 is on each solid arrow in (b) and (c), 1/12 and 1/4, are, respectively, on thin and thick arrows in (d).

Ø. For notational simplicity, however, we treat these common elements differently in set A. Thus, the cardinality of A is always $m+n$. Based on such definitions, as shown in Fig. 7, we investigate several approaches to globally integrating local distances and make further comparisons between them.

1) Option-1: Min NN: To match the common parts of two manifolds, a simple but intuitive method is to measure the similarity between their best suited local models, as illustrated in Fig. 7(a). It means to compute the minimum distance among subspace pairs in set A. Formally, it is defined:

$$d_1(\mathcal{M}_1,\mathcal{M}_2) = \min_{C_i\in\mathcal{M}_1}\min_{C'_j\in\mathcal{M}_2} d(C_i,C'_j) = \min\left\{d(C_i,C'_{N(i)})|_{i=1}^{m}, d(C_{N'(j)},C'_j)|_{j=1}^{n}\right\} \tag{19}$$

This option was exploited in our preliminary study [22] as well as the related work [2], where impressive results have been reported. However, as it only imposes weight 1 on the closest pair of local models, the integration result could be unstable and easily affected by outliers in the presence of noise.

2) Option-2: Mean NN: To incorporate more information from multiple NN pairs, it is easy to reach another integration way that simply computes the mean distance of all subspace pairs in A:

$$d_2(\mathcal{M}_1,\mathcal{M}_2) = \frac{1}{m+n}\left(\sum_{i=1}^{m} d\left(C_i,C'_{N(i)}\right) + \sum_{j=1}^{n} d\left(C_{N'(j)},C'_j\right)\right). \tag{20}$$

While the equal weight setting is quite straightforward, this option does not take account of global data distribution. More specifically, as shown in Fig. 7(b), it is unreasonable to treat all NN pairs of local models equally for large variations.

3) Option-3: Mean NN's NN (N^4): A more general intuition is that the *smaller* the distance of the pair is, the *larger* its weight should be. This motivates our third option to transfer some weights from the further pairs to those closer ones. Fig. 7(c) demonstrates the idea.

Specifically, for each C_i $(i=1,2,\ldots,m)$ in $\mathcal{M}_1$, we find its NN $C'_{N(i)}$ in $\mathcal{M}_2$. Then for $C'_{N(i)}$, we find inversely its NN $C_{N'(N(i))}$ in $\mathcal{M}_1$. Here, we call $C_{N'(N(i))}$ as the NN's NN (N^4) of C_i. By replacing the term $(C_i, C'_{N(i)})$ in (20) with $(C_{N'(N(i))}, C'_{N(i)})$, we then transfer the weight of the former pair to the latter one. In the same manner, for each C'_j $(j=1,2,\ldots,n)$ in $\mathcal{M}_2$, we can find its NN $C_{N'(j)}$ and N^4 $C'_{N(N'(j))}$ respectively. Then the term $(C_{N'(j)}, C'_j)$ in (20) is replaced with $(C_{N'(j)}, C'_{N(N'(j))})$. Finally, we derive a more reasonable definition of MMD in the following:

$$d_3(\mathcal{M}_1,\mathcal{M}_2) = \frac{1}{m+n}\left(\sum_{i=1}^{m} d\left(C_{N'(N(i))},C'_{N(i)}\right) + \sum_{j=1}^{n} d\left(C_{N'(j)},C'_{N(N'(j))}\right)\right). \tag{21}$$

For more intuitive illustration, let us see the two manifolds in Fig. 7(c). For the local model C_1 (i.e., $i=1$), its NN and N^4 are $C'_{N(1)} = C'_2$ and $C_{N'(N(1))} = C_2$ respectively. Therefore, the weight of the further pair (C_1, C'_2) (*red dashed arrow* in the figure) will be transferred to the closer pair (C_2, C'_2) (*red solid arrow*). Likewise, for the local model C'_1 (i.e., $j=1$), the weight of (C_2, C'_1) (*blue dashed arrow*) will also be transferred to (C_2, C'_2) (*blue solid arrow*).

We can see that option-3 combines the merits of option-1 and 2. Compared with option-1, it can guarantee more stable results by using information from more data. Compared with option-2, it can adaptively adjust the weights on different NN pairs in accordance with the real data characteristics more reliably.

4) Option-4: Earth Mover's Distance (EMD): Our fourth option exploits Earth Mover's Distance (EMD) [31], which shows promising performance in many applications. In this option, we apply EMD to compute the weights f_{ij} in (6). Two manifolds $\mathcal{M}_1$ and $\mathcal{M}_2$ are represented as two signatures: $\mathcal{M}_1 = \{(C_1, w_{C_1}), \ldots, (C_m, w_{C_m})\}$, $\mathcal{M}_2 = \{(C'_1, w_{C'_1}), \ldots, (C'_n, w_{C'_n})\}$, where w_{C_i} and $w_{C'_j}$ are the weights of the clusters C_i and C'_j. The weights $w_{C_i}/w_{C'_j}$ are used as the total supply of suppliers (i.e. $\mathcal{M}_1$) and the total capacity of consumers (i.e. $\mathcal{M}_2$) in EMD, with the default normalized value of $1/m$ and $1/n$ respectively. The EMD between $\mathcal{M}_1$ and $\mathcal{M}_2$ is computed by

$$d_4(\mathcal{M}_1,\mathcal{M}_2) = \sum_{i=1}^{m}\sum_{j=1}^{n}\hat{f}_{ij}d_{ij} \Big/ \sum_{i=1}^{m}\sum_{j=1}^{n}\hat{f}_{ij}, \tag{22}$$

where the SSD $d_{ij} = d(C_i, C'_j)$ is called ground distance, $\hat{f}_{ij}$ is the optimal flow that can be determined by solving the

following Linear Programming problem [31]:

$$\begin{aligned}&\hat{f}_{ij} = \arg\min_{f_{ij}} \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} d_{ij}\\ &\text{s.t. } \sum_{j=1}^{n} f_{ij} \le w_{C_i}, 1 \le i \le m; \quad \sum_{i=1}^{m} f_{ij} \le w_{C'_j}, 1 \le j \le n;\\ &f_{ij} \ge 0; \quad \sum_{i=1}^{m}\sum_{j=1}^{n} f_{ij} = \min\left(\sum_{i=1}^{m} w_{C_i}, \sum_{j=1}^{n} w_{C'_j}\right).\end{aligned} \tag{23}$$

In (23), the first two constraints limit the amount of supplies that can be sent by the clusters in $\mathcal{M}_1$ to their weights, and the clusters in $\mathcal{M}_2$ to receive no more supplies than their weights; the third constraint allows moving "supplies" from $\mathcal{M}_1$ to $\mathcal{M}_2$, not vice versa; the last constraint forces to move the maximum amount of supplies possible, named the total flow [31].

$\hat{f}_{ij}$ can be interpreted as the optimal match among local models from the two manifolds. Due to the "supply-consumption balance" constraint in (23), the EMD match will inevitably place some weights on those non-NN local model pairs, as shown in Fig. 7(d). The same problem also applies to other possible EMD weights, e.g., unit weights.

5) Comparisons of the Four Options: From above introduction, option-1 and option-3 conform to our intuition to impose more emphasis on those closer NN pairs. It is also interesting to note that under these two options, the MMD in (6) is consistent with the definition of SMD in (5) when a manifold has only one local model, i.e., $m = 1$ or $n = 1$.

As for computational cost, option-4 is the most expensive. Suppose the numbers of local models in two manifolds are the same, i.e., $m = n$, then its complexity is $O(m^3 \log(m))$ [31]. In contrast, the other three options are all of $O(m^2)$.

Besides option-1 to 4, other choices may also be explored. For example, if an option takes the form of "*Max NN*", then it readily reduces to the well-known Hausdorff distance [32]. However, for the FRIS task, the above four options are believed to be the most appropriate ones.

IV. Discussion

A. Comparisons With Related Work

From the view of set matching problem, MMD bears some resemblance to two representative methods, MSM [14] and "LLE + K-means" [17]. For *set modeling*, MSM represents an image set with a linear subspace, while "LLE + K-means" models it as a nonlinear manifold. However, the latter expresses the manifold with some pre-specified number of local models. On the contrary, MMD is able to extract multi-scale local models from the manifold adaptively and the number of local models for each manifold can be estimated using a measurable nonlinearity score in real-time for testing data. In the aspect of *SSD measure*, MSM and "LLE + K-means" respectively exploit the variation and exemplar based measures, while MMD reasonably fuses the two measures. In brief, MMD effectively combines the merits of the two methods and provides a general set similarity measure.

To address another different problem of measuring invariant image similarity, several methods were proposed in the literature, e.g., Joint Manifold Distance (JMD) [33] as well as related work [34], Multiresolution Manifold Distance (MRMD) [35], and Manifold Distance using the difference of convex functions (MDDC) [36]. Though their titles seem similar to MMD, their intrinsic properties are quite different from ours mainly in the following two aspects:

1) These methods mainly serve as distance measures between *images* to achieve invariance to parameterized image transformations, following the earlier work of tangent distance (TD) [37]. Whereas, MMD aims to measure the similarity between two sets of images from the nonparametric viewpoint.
2) The notion of "manifold distance" in these methods means the distance from a reference point to the transformation manifold [36], and it is actually defined between points in linear subspaces. On the contrary, MMD formulates the distance of data variations on general manifolds.

B. Complexity Analysis

The computational complexity of MMD is basically dominated by the following four parts.

(1) Constructing local models (i.e., MLPs) based on L-HDC. To compute the three $N \times N$ matrices $\boldsymbol{D}_E$, $\boldsymbol{D}_G$ and $\boldsymbol{R}$ in advance, using Dijkstra's algorithm with Fibonacci heaps, the complexity is $O(N^2 \log N)$ [28], where N is the number of samples in the image set. Then the complexity of Algo.1 mainly relies on step 2.4 to compute the score $\beta_l^{(i)}$ and $\beta_r^{(i)}$ according to Eq.(8). For simplicity, we assume the two child clusters $\boldsymbol{X}_l^{(i)}$ and $\boldsymbol{X}_r^{(i)}$ are of equal size. Then to obtain the complete clustering hierarchy, the complexity of Algo.1 is

$$O\left(\sum_{p=1}^{\lfloor \log N \rfloor} (2^p (N/2^p)^2)\right) \approx O(N^2).$$

(2) Computing PCA subspaces for local models. For each local model $C_i (i = 1, 2, \ldots, m)$, its data matrix is of size $D \times N_i$. Assume the local models are of equal size, then $N_i \approx N/m$. The PCA computation mainly involves eigenvalue decomposition of the $D \times D$ covariance matrix. Since it is often the case that $D > N > N_i$, the eigen-decomposition can be conducted on an $N_i \times N_i$ matrix plus some low complexity matrix multiplications. Thus the complexity of this part is about $O((N/m)^3 \cdot m)$.

(3) Computing local model distances (i.e., SSD) based on principal angles. Suppose the numbers of local models in two manifolds are the same, then we need $m \times m$ times of principal angles computation. With the SVD algorithm in Eq. (10), the total complexity is thus $O(d^3 \cdot m^2)$, where d is the PCA subspace dimension of the local model.

(4) Integrating the local distances. As discussed in Section III-C-5, adopting our option-3, the complexity is $O(m^2)$.

To sum up, the total complexity of MMD is the sum of the above four parts and can be roughly approximated by $O(N^3)$.

V. Experimental Results

The proposed method is evaluated in the FRIS task, where both gallery and probe image sets are modeled as manifolds, and identification is achieved by seeking the minimum MMD.

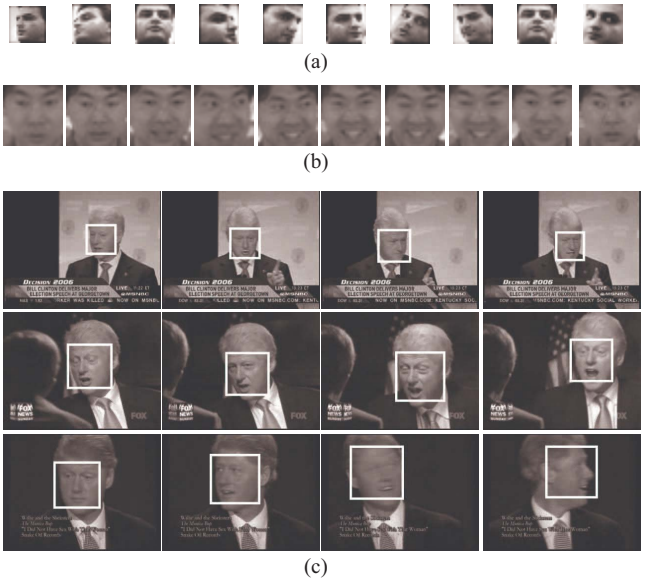

Fig. 8. Examples of three face databases. Each row contains representative facial images from one video clip. (a) Honda/UCSD. (b) CMU MoBo. (c) YouTube Celebrities, where each of the three rows comes from a different session, and the original video frames with automatic face detection are shown.

A. Databases Description and Settings

We consider three datasets with different characteristics, including two benchmark datasets: Honda/UCSD [8], CMU MoBo [38], and one much challenging dataset: YouTube Celebrities [7], to ensure extensive evaluations of different methods. The Honda/UCSD consists of 59 video sequences involving 20 persons. Each video contains about 400 frames covering large variations in head movement as well as in facial expression. The CMU MoBo contains 96 sequences of 24 subjects. Each subject has 4 sequences captured in different walking situations: holding a ball, fast walking, slow walking, and walking on incline. Each sequence has about 300 frames. The YouTube dataset contains 1,910 video clips of 47 subjects collected from YouTube. Each person has on average a total of 41 clips, which are divided into 3 sessions taken at different times and scenes. Each clip contains hundreds of frames, which are mostly low resolution and highly compressed. For all three databases, we used a cascaded face detector [39] to collect faces in each video. Faces in Honda/UCSD were resized to 20×20 gray images, and those in CMU MoBo and YouTube were resized to 30×30. Histogram equalization was employed to eliminate lighting effects. Some examples are shown in Fig. 8.

On all datasets, we conducted ten-fold cross validations, i.e., 10 randomly selected training and testing combinations of the video clips, for reporting identification rates. For both Honda and MoBo, each person had one clip for training and the rest for testing. For YouTube, in each of the ten-fold cross validations, one person had a total of nine randomly chosen clips from his/her three sessions (i.e., three clips from each session) for experiment, where three clips were used for training and six for testing. For a relatively easy setting, one of the three clips from each session was for training and the rest two for testing.

B. Comparative Methods and Parameter Settings

We compared the performance of the following methods:

1) Nearest Neighbor matching in LLE + K-means clustering [17], which is a typical exemplar-based method,
2) Mutual Subspace Method (MSM) [14], which is a typical variation-based method,
3) Kernel Principal Angles (KPA) [16][1] and Kernel Grassmannian distances (KGD) [12], which are two representative nonlinear extensions of principal angles,
4) Constrained MSM (CMSM) [5] and Discriminant Canonical Correlations (DCC) [3][2], which are two representative discriminant methods over sets,
5) The proposed MMD method.

In our experiments, to compare different methods, their important parameters were tuned empirically within a wide range. In LLE + K-means, we used the same parameter setting as [17]. For each training video sequence, $k = 5$ exemplars were extracted, and the identity of the probe image set was determined using majority voting scheme. In KPA, a sixth order monomial expansion kernel and the first 20 principal angles were used as [16]. In KGD, we exploited the Chordal distance (i.e. the projection metric in (14)) and chose Gaussian function with bandwidth $\sigma = 2$ as [12] to compute the kernel subspace.

For MSM/CMSM/DCC, we followed the evaluation settings similar to [3]. PCA was first performed to learn the linear subspace of each image set, and the subspace dimension was around 15 by preserving 95% energy. Then, the dimensions of discriminative subspaces in CMSM and DCC were determined optimally in terms of identification rate for each database. Finally, to measure the set similarity, both CMSM and DCC exploited all canonical correlations. For MSM, we reported its best possible results by tuning the number of canonical correlations. For the DCC learning on Honda and MoBo, the single training image set from each class was randomly divided into two subsets to construct the within-class sets as in [3].

For MMD, we adopted the formulation in (21), i.e., option-3 (*Mean* N^4) for the global integration. First, we need to set the number of local models for each manifold, i.e. m, n. Its appropriate value can be determined from the nonlinearity score curve, as shown in Fig. 9. For instance, we can set it to a value whose first-order derivative approximates zero. Typical empirical value for one set with $300 \sim 400$ images is $5 \sim 8$. Second, for representing the local model C_i, we set the PCA dimension d_i by preserving 95% variance, which was around 5. Finally, the formal SSD in (16) was used with all principal angles.

C. Comparison Results and Analysis

Table I summarizes all the comparison results. Each reported rate is an average over the ten runs of cross validation. We next highlight some observations on the results.

First, the proposed MMD consistently outperforms the two baseline non-discriminative methods, i.e., LLE + K-means

[1] We used the original authors' implementation.

[2] For MSM/CMSM/DCC, we used the implementations all shared by the authors of DCC [3].

TABLE I
EVALUATION RESULTS OBTAINED BY TEN-FOLD CROSS VALIDATION ON DIFFERENT DATA SETS

Dataset	Mean and standard deviation of recognition rates of different methods						
	LLE + K-means	MSM	KPA	KGD	CMSM	DCC	MMD
Honda/UCSD	0.894 ± 0.03	0.925 ± 0.04	0.953 ± 0.04	0.944 ± 0.04	0.975 ± 0.02	0.980 ± 0.01	0.971 ± 0.02
CMU MoBo	0.879 ± 0.04	0.899 ± 0.03	0.905 ± 0.04	0.912 ± 0.05	0.924 ± 0.04	0.919 ± 0.05	0.935 ± 0.02
YouTube	0.592 ± 0.06	0.608 ± 0.04	0.633 ± 0.05	0.620 ± 0.05	0.647 ± 0.03	0.673 ± 0.03	0.652 ± 0.03

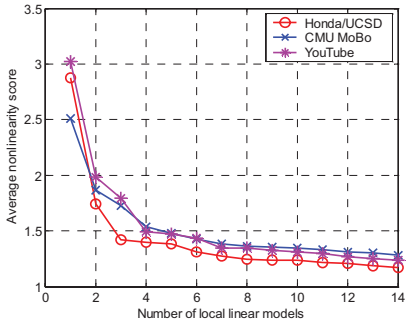

Fig. 9. Average nonlinearity score curves, each of which corresponds to an example image set from the three face databases.

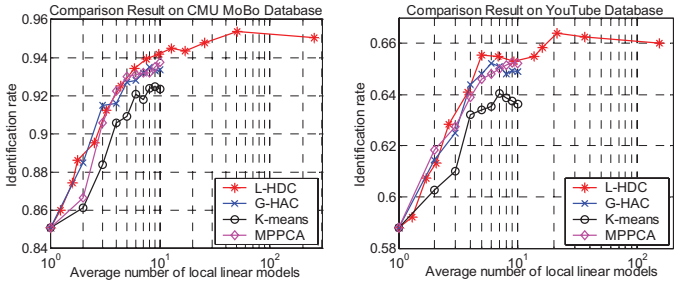

Fig. 10. Comparison of different clustering methods.

and MSM, by a large margin. As discussed in Section IV-A, MMD reasonably integrates the preferable features of them. Among the three methods, LLE + K-means gives the lowest recognition rates though it exploits manifold to extract training exemplars. This is mainly because its testing scheme is somewhat simple by classifying samples in each testing image set separately and this method can not fully model the variability of the image set [4].

Second, by using principal angles and modeling data nonlinearity *implicitly* via kernel trick, the two methods KPA and KGD deliver similar performance, as reported in [12], and have improved MSM with an accuracy increase of about 3%. However, they are still inferior to our MMD that uses a collection of local linear subspaces to *explicitly* characterize the nonlinear manifold. Kernel based methods generally have the difficulty to select appropriate kernel functions and tune the parameters. This is reflected partially from their relatively higher standard deviations on all the databases. In addition, their prohibitive computational burden also makes them less appealing, as will be demonstrated in the next section.

Third, compared with the two discriminant methods, CMSM and DCC, our non-discriminative MMD yields comparable result, which demonstrates the potential of MMD as a general distance for various applications. The superiority of DCC and CMSM is not surprising since they have effectively exploited the discriminative information from training data while all other methods do not involve the training phase. Note that in CMU MoBo, MMD even delivers the highest recognition rate impressively. We attribute this to our better manifold modeling method, and more importantly the SSD definition by fusing the variation and exemplar based measures in a more principled manner, which will also be further confirmed in the next section.

D. Evaluations of the Different Ingredients of MMD

We conducted further experiments to investigate the three ingredients of MMD discussed in Section III. We also performed an experimental study to address a practical problem of noisy set data. In the last, we experimentally compared the computational cost of our MMD with other recognition approaches. The two challenging datasets MoBo and YouTube were used for these evaluations.

1) Local Linear Model Construction: We compared our L-HDC with three alternative clustering methods exploited in previous literatures: HAC [4], K-means [17] and mixtures of probabilistic PCA (MPPCA) [2]. In [4], the distance between two clusters was measured by geodesic distance and we call it G-HAC. In K-means, Euclidean distance was used. In MPPCA, as the setting of [2], dimensionality reduction (from 900D to 150D) using PCA was performed to the original image data before the estimation of MPPCA model.

In our experiment, L-HDC was first conducted on each image set to obtain the complete clustering hierarchy. The MMD between two image sets was then computed by using their respective clustering level associated with the same nonlinearity score. Fig. 10 shows the comparison results under varying scores. As shown in Fig. 9, the number of clusters gives an intuitive reflection of the nonlinearity score, and different image sets may produce different numbers of clusters under the same score. Hence, the average number of local models for L-HDC in Fig. 10 is not necessarily integer value. In our testing, the average number of clusters (i.e. local models) in the last clustering level for MoBo/YouTube is 50.66/36.40 respectively, shown as the last but one point in the L-HDC curve. Moreover, we tested an extreme case by treating each single image sample as one local model and performing the "min NN" matching for recognition. In fact, this is exactly the method of [10], which we call Single Sample Matching (SSM) in the following. In this case, the average number is 255.87/153.11 for MoBo/YouTube respectively, shown as the last point in the L-HDC curve. For comparison, the other three clustering methods were also performed on each image set to obtain multi-level clusters, which were then similarly used to compute MMD under different clustering levels as done in L-HDC. However, unlike L-HDC, these three methods utilized

TABLE II
COMPUTATION TIME (SECONDS) OF DIFFERENT CLUSTERING METHODS

Algorithm	Number of local linear models				
	2	6	10	14	18
L-HDC	0.142	0.180	0.196	0.212	0.236
G-HAC	59.552	59.451	59.319	59.252	59.169
K-means	0.765	2.893	3.396	3.927	4.487
MPPCA	1.237	3.219	5.280	7.742	10.155

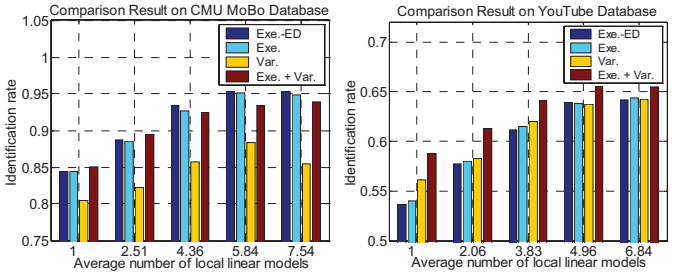

Fig. 11. Comparison of different SSD measures.

the same number of clusters for all image sets in each level of MMD computation.

From Fig. 10, we observe that with increased number of local models, all of the four methods enjoy large performance gains, indicating the contribution of our local models representation of the image set manifold. As the number of local models exceeds some point (usually less than 10), the accuracy correspondingly has only marginal changes with added local models. It is also interesting to find that in the extreme case, SSM can yield very close performance to the best of MMD, however, at much higher computational cost. As also discussed in Section I-A, this method is much more sensitive to the effect of outliers. These will be experimented in the following section.

While comparing the recognition accuracy by using the four clustering methods, it can be seen that L-HDC and G-HAC, both exploiting geodesic distance, similarly deliver superior performances to K-means, which further confirms the nonlinear manifold structure of the image sets. Combining with more sophisticated techniques, MPPCA also outperforms K-means and gives similar accuracy to the other two. In addition, the gain of L-HDC over G-HAC is mainly attributed to two factors. One is its explicit linearity guarantee of the clusters, and the other is its adaptive selection of the number of clusters, rather than a universal setting for different image sets in G-HAC.

Besides the accuracy, the computation time of the four methods is also compared, as shown in Table II. The results were obtained by averaging the time of 20 runs of each algorithm on 20 image sets, each with 300 images. Note that, the time for L-HDC has included the computation of the distance matrices $\boldsymbol{D}_E$ and $\boldsymbol{D}_G$. From Table II, L-HDC is shown to be the most efficient with a significant margin over G-HAC. Also, we observe that with the same increase of cluster number, the time increment for L-HDC/G-HAC is much less than that of K-means. This is because both L-HDC and G-HAC need to compute point-pair distances (the most expensive part) only once and their iterative steps mainly involve simple splitting or merging operations, while K-means needs to compute point-pair distances again in each new iteration. Furthermore, compared with K-means, MPPCA still seems more time demanding, though a PCA preprocessing has been conducted for significant reduction in computation as [2], making it less appealing in efficiency compared with our method.

2) Local Model Distance Measure: In this experiment, we compared the single exemplar-based SSD in (15) and variation-based SSD in (14) with our formal SSD in (16), referred to as "Exe.", "Var." and "Exe. + Var.", respectively. We also compared "Exe." with the exemplar distance defined in (1), referred to as "Exe.-ED". The comparison results are plotted in Fig. 11. From the figure, we have the following observations.

(1) There is no single distance that is universally optimal for all databases. In MoBo, "Exe." outperforms "Var." noticeably by more than 5%. One possible reason is that two sequences from different persons but with the same walking situation could have high correlation, making their variation-based SSD relatively small. On the contrary, "Var." yields slightly better result than "Exe." in YouTube. This can be also due to the database itself. See the examples in Fig. 8(c), for different clips of one person captured at different times and scenes, their appearance is likely to undergo considerably large changes, whereas their common variation modes (e.g., the person strikes a certain pose or expression) can remain relatively stable across different scenarios. In this case, the variation-based SSD is more reliable than the exemplar-based SSD.

(2) The formal SSD "Exe. + Var." generally outperforms the individual distances, especially in YouTube, demonstrating that combining the two single distances in a principled way is the best solution in practice. This also validates the rationality and effectiveness of the weighting scheme in (16) by imposing the factor $1/r$ to balance the two distance measures. Furthermore, the two exemplar distances, "Exe." and "Exe.-ED" deliver similar performances on both testing databases. This shows the feasibility of using the correlation-based distance in (15) as an alternative to the PPD in (1) for deriving our formal SSD.

3) Global Integration of Local Distances: We first verified the property of the four integration options when combined with different SSD measures. As mentioned in Section III-C-4, the EMD option has two types of weight settings, i.e., the normalized and unit weights, denoted as "EMD-Norm." and "EMD-Unit" respectively. The number of local models in MMD was fixed at 7.54 and 6.84 for MoBo and YouTube respectively (corresponding to the rightmost setting in Fig. 11). From the result in Fig. 12, it can be seen the four options demonstrate consistent contrast across different SSD measures. By emphasizing those closer NN local model pairs, option-1 (*Min NN*) and option-3 (*Mean* N^4) generally exhibit similar characteristics and outperform option-2 (*Mean NN*) and option-4 (*EMD*). In contrast, option-4 with either normalized or unit weights treats all local models in each manifold equally and delivers the worst performance. These observations all conform to the theoretical analyses in Section III-C.

We further evaluated the resistance of different options to a common problem in set-based recognition, i.e., unbalanced set

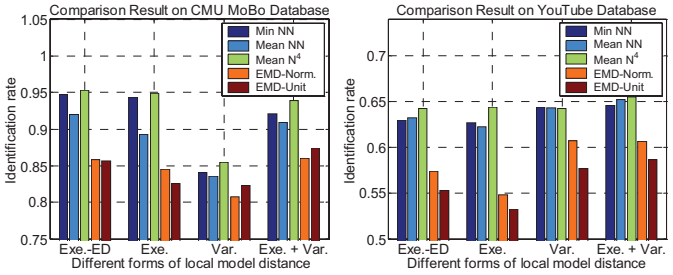

Fig. 12. Comparison of different options for integrating local distances.

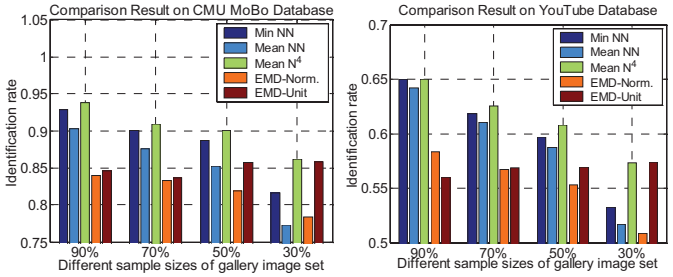

Fig. 13. Effect of unbalanced set size on MMD global integration options.

size, which will inevitably cause those bigger gallery sets to yield closer similarity to a probe set. We simulated the problem by down-sampling the gallery image sets while keeping the probe sets with the original size. Since our image sets are all from videos, one possible way is to remove some sub-clips with consecutive images from the original video. Specifically, each gallery set was first divided into 10 sub-clips of equal length. Then a reduced-size version of the set was produced by randomly removing certain number of the sub-clips. We have tested four versions by keeping 90%, 70%, 50% and 30% samples in the reduced gallery set. For set matching, we also implemented a scheme to simulate the practical scenario: once a comparison between a probe set and a gallery set is to be conducted, if they belong to the same class then we take the reduced gallery version to compute their similarity; otherwise we take the original size gallery set.

In general, with the set size decreasing, the number of resulted local models also drops accordingly. From the comparative results in Fig. 13, one can find that the four options, except for "EMD-Unit", all consistently have their performance deteriorated with the degenerated gallery sets. The superiority of option-3 is further validated from the contrast. Especially in the hardest case (30%), option-1 becomes obviously inferior to option-3. It is also reasonable to see the accuracy raise of "EMD-Unit", since the unbalanced number of local models between the gallery and probe image sets make this option focus on matching the closest part of the two sets favorably.

4) Problem of Noisy Set Data: Another common challenge in real-world application is that the image sets contain noisy data (i.e., images outside the category), as noted in [40]. In this section, we experimentally studied this problem and compared MMD with other three principal angles-based methods (MSM, CMSM and DCC) as well as the Single Sample Matching (SSM) method. We used the similar setting in [40] and

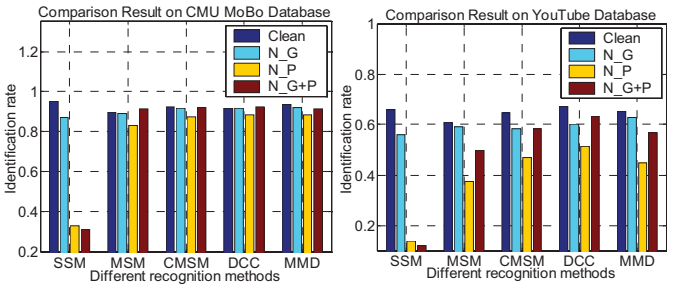

Fig. 14. Effect of noisy set data on different recognition methods.

conducted three experiments in which the gallery and/or the probe sets were systematically corrupted by adding one image from each of the other classes. The three cases are referred to as "N_G" (only gallery has noise), "N_P" (only probe has noise), and "N_G+P" (both).

From the result in Fig. 14, it is observed that along with other principal angles-based methods, our MMD shows relatively slight performance drops and is much more robust than SSM against the noisy data. This can be mainly attributed to the subspace based modeling and matching since it is a statistic of samples and the noisy set samples are largely filtered out with an average filter during subspace computation. In contrast, based on matching the closest sample, SSM relies highly on the location of each individual sample and can be heavily deteriorated by outliers, as mentioned in Section I-A.

5) Computational Cost Evaluation: Since time efficiency is an important concern for set classification tasks, here we make an experimental comparison of the computational cost of different recognition methods. LLE + K-means was excluded from this experiment for its rather different classification scheme from the others. Two image sets from the MoBo database, each with about 300 images, were selected to report the result on a Pentium IV, 2.8 GHz PC with 2 GB of RAM.

For MMD, as discussed in Section IV-B, its complexity is dominated by four parts, where the first two parts are mainly for set modeling and the last two for set matching. By modeling the image set with a typical number of 6 local subspaces, the time cost for each part is tabulated in Table III. One can see that the first part to construct MLPs is the most expensive, which mainly involves the computation of geodesic distances. For MSM/CMSM/DCC, they share the same set modeling by computing a single PCA for the image set, while the matching phase of MSM obviates the discriminant projection in CMSM/DCC. For SSM/KPA/KGD, they all directly conduct set matching without explicit set modeling. The heavy computational burden makes them less appealing. While the efficiency of SSM seems insignificantly inferior to MMD in terms of total time cost, it is worth noting that in real applications, their efficiency difference will mainly rely on the matching phase where MMD is about 43 times faster than SSM. This is because the set models only need to be constructed once but can be used repeatedly, e.g., in need of comparing a probe set with multiple gallery sets in FRIS.

TABLE III
COMPARISON OF COMPUTATION TIME (SECONDS) FOR DIFFERENT SET CLASSIFICATION METHODS

Algorithm	Modeling		Matching		Total
	MLP	PCA	SSD	Integ.*	
MMD	0.1795	0.1109	0.0283	0.0001	0.3188
SSM	N/A		1.2244	0.0012	1.2256
MSM	0.3127		0.0006		0.3133
CMSM/DCC	0.3127		0.0071		0.3198
KPA	N/A		8.4467		8.4467
KGD	N/A		7.7307		7.7307

*Integration.

VI. CONCLUSION

Recognizing sets of images undergoing large variations is a challenging problem. By representing each image set as a manifold, the problem is formulated as measuring the distance between manifolds. We propose a general framework of Manifold to Manifold Distance (MMD), and present several technical contributions for its computation. Extensive experiments on Face Recognition with Image Sets (FRIS) demonstrate that the proposed method consistently outperforms other competing methods, and is also promisingly comparable to the state-of-the-art discriminant methods over sets.

Currently the MMD is exploited mainly as a general set similarity measure. In the future, we intend to incorporate the video temporal dynamic features. Another interesting direction could be to explore the most useful information involved in principal angles, like the learning approach in CMSM/DCC, and measure the local model distance in more sophisticated manner. In addition, it is also beneficial to extend the work in order to accommodate the cases of data with critical sparsity, which would deteriorate the manifold assumption and representation in the method.

ACKNOWLEDGMENT

The authors would like to thank Dr. T-K. Kim and Dr. L. Wolf for sharing the codes of DCC/CMSM/MSM and KPA, respectively.

REFERENCES

[1] O. Arandjelović, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell, "Face recognition with image sets using manifold density divergence," in *Proc. Comput. Vision Pattern Recog. Conf.*, Jun. 2005, pp. 581–588.

[2] T-K. Kim, O. Arandjelović, and R. Cipolla, "Boosted manifold principal angles for image set-based recognition," *Pattern Recog.*, vol. 40, no. 9, pp. 2475–2484, 2007.

[3] T-K. Kim, J. Kittler, and R. Cipolla, "Discriminative learning and recognition of image set classes using canonical correlations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 6, pp. 1005–1018, Jun. 2007.

[4] W. Fan and D.-Y. Yeung, "Locally linear models on face appearance manifolds with application to dual-subspace based classification," in *Proc. IEEE Comput. Vision Pattern Recog. Conf.*, 2006, pp. 1384–1390.

[5] K. Fukui and O. Yamaguchi, "Face recognition using multi-viewpoint patterns for robot vision," in *Proc. Int. Symp. Robot. Res.*, 2003, pp. 192–201.

[6] T-K. Kim, J. Kittler, and R. Cipolla, "On-line learning of mutually orthogonal subspaces for face recognition by image sets," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1067–1074, Apr. 2010.

[7] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley, "Face tracking and recognition with visual constraints in real-world videos," in *Proc. Comput. Vision Pattern Recog. Conf.*, Jun. 2008, pp. 1–8.

[8] K. Lee, M. H. Yang, and D. Kriegman, "Video-based face recognition using probabilistic appearance manifolds," in *Proc. Comput. Vision Pattern Recog. Conf.*, 2003, pp. 313–320.

[9] X. Liu and T. Chen, "Video-based face recognition using adaptive hidden Markov models," in *Proc. Comput. Vision Pattern Recog. Conf.*, 2003, pp. 340–345.

[10] S. Satoh, "Comparative evaluation of face sequence matching for content-based video access," in *Proc. Int. Conf. Autom. Face Gesture Recog.*, 2000, pp. 163–168.

[11] G. Shakhnarovich, J. W. Fisher, and T. Darrell, "Face recognition from long-term observations," in *Proc. Eur. Conf. Comput. Vision*, 2002, pp. 851–868.

[12] T. Wang and P. Shi, "Kernel grassmannian distances and discriminant analysis for face recognition from image sets," *Pattern Recog. Lett.*, vol. 30, no. 13, pp. 1161–1165, 2009.

[13] S. Zhou, V. Krüeger, and R. Chellappa, "Probabilistic recognition of human faces from video," *Comput. Vision Image Underst.*, vol. 91, no. 1, pp. 214–245, 2003.

[14] O. Yamaguchi, K. Fukui, and K. Maeda, "Face recognition using temporal image sequence," in *Proc. Int. Conf. Autom. Face Gesture Recog.*, 1998, pp. 318–323.

[15] J. Hamm and D. D. Lee, "Grassmann discriminant analysis: A unifying view on subspace-based learning," in *Proc. Int. Conf. Mach. Learn.*, 2008, pp. 376–383.

[16] L. Wolf and A. Shashua, "Learning over sets using kernel principal angles," *J. Mach. Learn. Res.*, vol. 4, no. 10, pp. 913–931, 2003.

[17] A. Hadid and M. Pietikäinen, "From still image to video-based face recognition: An experimental analysis," in *Proc. IEEE 6th Int. Conf. Autom. Face Gesture Recog.*, May 2004, pp. 813–818.

[18] Å. Björck and G. H. Golub, "Numerical methods for computing angles between linear subspaces," *Math. Comput.*, vol. 27, no. 123, pp. 579–594, 1973.

[19] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 34, pp. 321–372, 1936.

[20] A. Edelman, T. A. Arias, and S. T. Smith, "The geometry of algorithms with orthogonality constraints," *SIAM J. Matrix Anal. Appl.*, vol. 20, no. 2, pp. 303–353, 1999.

[21] P. Belhumeur and D. Kriegman, "What is the set of images of an object under all possible illumination conditions?" *Int. J. Comput. Vision*, vol. 28, no. 3, pp. 245–260, Jul. 1998.

[22] R. Wang, S. Shan, X. Chen, and W. Gao, "Manifold-manifold distance with application to face recognition based on image set," in *Proc. Comput. Vision Pattern Recog. Conf.*, 2008, pp. 2940–2947.

[23] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. Comput. Vision Pattern Recog. Conf.*, 1991, pp. 586–591.

[24] L. Wang, X. Wang, and J. Feng, "Subspace distance analysis with application to adaptive Bayesian algorithm for face recognition," *Pattern Recog.*, vol. 39, no. 3, pp. 456–464, 2006.

[25] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 22, pp. 2323–2326, Dec. 2000.

[26] R. Wang and X. Chen, "Manifold discriminant analysis," in *Proc. Comput. Vision Pattern Recog. Conf.*, 2009, pp. 429–436.

[27] R. Wang, S. Shan, X. Chen, J. Chen, and W. Gao, "Maximal linear embedding for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1776–1792, Sep. 2011.

[28] J. Tenenbaum, V. Silva, and J. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 22, pp. 2319–2323, Dec. 2000.

[29] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via local tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, 2004.

[30] A. M. Martínez, "Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 6, pp. 748–763, Jun. 2002.

[31] Y. Rubner, C. Tomasi, and L. Guibas, "The earth Mover's distance as a metric for image retrieval," *Int. J. Comput. Vision*, vol. 40, no. 2, pp. 99–121, 2000.

[32] M. P. Dubuisson and A. K. Jain, "A modified hausdorff distance for object matching," in *Proc. 12th Int. Conf. Pattern Recog.*, vol. 1. Oct. 1994, pp. 566–568.

[33] A. W. Fitzgibbon and A. Zisserman, "Joint manifold distance: A new approach to appearance based clustering," in *Proc. Comput. Vision Pattern Recog. Conf.*, 2003, pp. 26–33.
[34] O. Arandjelović and A. Zisserman, "Automatic face recognition for film character retrieval in feature-length films," in *Proc. IEEE Comput. Vision Pattern Recog. Conf.*, Jun. 2005, pp. 860–867.
[35] N. Vasconcelos and A. Lippman, "A multiresolution manifold distance for invariant image similarity," *IEEE Trans. Multimedia*, vol. 7, no. 1, pp. 127–142, Feb. 2005.
[36] E. Kokiopoulou and P. Frossard, "Minimum distance between pattern transformation manifolds: Algorithm and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 7, pp. 1225–1238, Jul. 2009.
[37] P. Simard, Y. L. Cun, J. Denker, and B. Victorri, "Transformation invariance in pattern recognition-tangent distance and tangent propagation," *Neural Netw. Tricks Trade*, LNCS 1524, pp. 239–274, 1998.
[38] R. Gross and J. Shi, "The CMU motion of body (MoBo) database," Robotics Institute, Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-RI-TR-01-18, Jun. 2001.
[39] P. Viola and M. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137–154, 2004.
[40] H. Cevikalp and B. Triggs, "Face recognition based on image sets," in *Proc. IEEE Comput. Vision Pattern Recog. Conf.*, Jun. 2010, pp. 2567–2573.

Ruiping Wang (S'08–M'11) received the B.S. degree in applied mathematics from Beijing Jiaotong University, Beijing, China, in 2003, and the Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, in 2010.

He has been a Post-Doctoral Researcher with the Department of Automation, Tsinghua University, Beijing, since July 2010. From 2010 to 2011, he was a Research Associate with the Computer Vision Laboratory, Institute for Advanced Computer Studies, University of Maryland, College Park. His current research interests include computer vision, pattern recognition, and machine learning.

Shiguang Shan (M'04) received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2004.

He has been with ICT, CAS, since 2002, where he has been a Full Professor, since 2010. He has authored or co-authored more than 120 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, and image processing. His current research includes face recognition-related topics.

Dr. Shan has served as an Area Chair for a number of international conferences, such as ICCV2011, ICPR2012, and ACCV2012. He has been on the editorial board of *Neurocomputing* since 2012. He was a recipient of the China's State Scientific and Technological Progress Award in 2005 for his work on face recognition technologies, and the recipient of the Best Student Poster Award Runner-Up in CVPR2008 and the Silver Medal of Scopus Future Star of Science Award in 2009.

Xilin Chen (M'00–SM'09) received the B.S., M.S., and Ph.D. degrees in computer science from the Harbin Institute of Technology, Harbin, China, in 1988, 1991, and 1994, respectively.

He was a Professor with the Harbin Institute of Technology from 1999 to 2005. He was a Visiting Scholar with Carnegie Mellon University, Pittsburgh, PA, from 2001 to 2004. He has been with the Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing, China, since August 2004. He is the Director of the Key Laboratory of Intelligent Information Processing, CAS. He has authored or co-authored one book and over 200 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He is an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING, an Area Editor of the *Journal of Computer Science and Technology*, and an Associate Editor of the *Chinese Journal of Computers*. He has served as a Program Committee Member for more than 30 conferences.

Dr. Chen was a recipient of several awards, including the China's State Scientific and Technological Progress Award in 2000, 2003, and 2005, for his research work.

Qionghai Dai (SM'05) received the B.S. degree from Shanxi Normal University, Xi'an, China, in 1987, and the M.E. and Ph.D. degrees from Northeastern University, Shenyang, China, in 1994 and 1996, respectively.

He has been with the Faculty of Tsinghua University, Beijing, China, since 1997. He is currently a Professor and the Director of the Broadband Networks and Digital Media Laboratory. His current research interests include video communication, computer visions, and computational photography.

Wen Gao (M'92–SM'05–F'09) received the Ph.D. degree in electronics engineering from the University of Tokyo, Tokyo, Japan, in 1991.

He is currently a Professor of computer science with Peking University, Beijing, China. He was a Professor with the Harbin Institute of Technology, Harbin, China, from 1991 to 1995, and a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, from 1996 to 2006. He has published extensively, including five books and over 600 technical articles in refereed journals and conference proceedings in the areas of image processing, video coding and communication, pattern recognition, multimedia information retrieval, multimodal interfaces, and bioinformatics.

Prof. Gao is a fellow of the Chinese Academy of Engineering. He serves on the editorial board for several journals, such as the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT, the *EURASIP Journal of Image Communications*, and the *Journal of Visual Communication and Image Representation*. He chaired a number of prestigious international conferences on multimedia and video signal processing, such as the IEEE International Conference on Multimedia and Expo and the Association of Computing Machinery Multimedia, and also served on the advisory and technical committees of numerous professional organizations.