Manifold Alignment via Corresponding Projections

Deming Zhai[12]
dmzhai@jdl.ac.cn

Bo Li[12]
bli@jdl.ac.cn

Hong Chang[23]
hchang@jdl.ac.cn

Shiguang Shan[23]
sgshan@jdl.ac.cn

Xilin Chen[23]
xlchen@jdl.ac.cn

Wen Gao[14]
wgao@jdl.ac.cn

[1] School of Computer Science and Technology,
Harbin Institute of Technology, China

[2] Digital Media Research Center,
Institute of Computing Technology,
CAS, China

[3] Key Laboratory of Intelligent Information Processing,
Chinese Academy of Sciences, China

[4] Institute of Digital Media,
Peking University, China

Abstract

In this paper, we propose a novel manifold alignment method by learning the underlying common manifold with supervision of corresponding data pairs from different observation sets. Different from the previous algorithms of semi-supervised manifold alignment, our method learns the explicit corresponding projections from each original observation space to the common embedding space everywhere. Benefiting from this property, our method could process new test data directly rather than re-alignment. Furthermore, our approach doesn't have any assumption on the data structures, thus it could handle more complex cases and get better results compared with previous work. In the proposed algorithm, manifold alignment is formulated as a minimization problem with proper constraints, which could be solved in an analytical manner with closed-form solution. Experimental results on pose manifold alignment of different objects and faces demonstrate the effectiveness of our proposed method.

1 Introduction

In many real-world applications, the same object (*e.g.*, pose, face) may have different observations (or descriptions) which are highly related but sometimes look different from each other, such as the videos of the same scene from different viewpoints, the image sequences of the same action for different objects, the video and audio segments that come from the same circumstances, and so on. How to find the correspondence between the data points of different observation sets is a hot topic. Due to the fact that different datasets might be located in different high-dimensional spaces and represented by different features, it is difficult to match the data in their original observation spaces. From the geometric perspective,

each observation set (*e.g.*, a sequence of face images under various poses) forms a manifold. Given that these observations are from the same object, it is reasonable to assume that some common features across different observation spaces can be represented in an underlying common manifold. The shared low-dimensional embeddings, as better descriptions of the intrinsic geometry and relationship between different manifolds, are expected to benefit the subsequent task of datasets alignment.

In order to obtain the intrinsic low-dimensional representations of samples, many algorithms of manifold learning have been proposed in the literature. Traditional techniques such as principle components analysis (PCA) have been extensively used for linear dimensionality reduction. Other proposed linear methods include locally preserving projection (LPP) [7], neighbors preserving embedding (NPE) [8], and so on. Whereafter, some nonlinear methods such as locally linear embedding (LLE) [12], ISOMAP [15], and Laplacian Eigenmaps [4, 5] have shown promising results in nonlinear dimensionality reduction problems. Note that most algorithms of manifold learning are to find the intrinsic low-dimensional embeddings of a given dataset from only one high-dimensional observation. In this paper, we focus on the task of finding the underlying common manifold of multiple observation datasets, *i.e.*, aligning different manifolds.

During the past several years, manifold alignment has attracted much attention in the community of machine learning and computer vision. In [10], a semi-supervised nonlinear manifold alignment (NMA) algorithm is proposed. Shon *et al.* [13] propose an algorithm based on Gaussian process regression to learn the shared latent structure between datasets and apply it in image synthesis and robotic imitation. In [6], the authors propose to align the data manifolds into the predefined target coordinates. Xiong *et al.* [17] propose a method of manifold alignment with loose semi-supervised priors and formulate manifold alignment as an energy optimization problem. However, all above alignment methods cannot process new test data without retraining. More recently, some researchers propose a new two-stage algorithm for manifold alignment based on Procrustes analysis [16], which can be generalized to new data points. Nevertheless, Procrustes analysis, which only learns single affine transformation between one manifold and others, has limitations on the strong assumption on data structures. In addition, Verbeek [18] propose a shared manifold method with the perspective of probability theory. The method, however, restricted to one-to-one mappings, and so cannot model multi-modal conditionals.

In this paper, we propose a novel manifold alignment method via corresponding projections under the semi-supervised learning setting. In our algorithm, manifold alignment is formulated as a minimization problem with constraints. Our optimization could be solved in an analytical manner with closed-form solution. The proposed method is more general in the following three senses: (1) It learns explicit corresponding mappings from different manifolds to the underlying common embeddings, thus overcome the limitation of most previous methods on out-of-sample extension [3] ability. (2) It defines the corresponding mappings for different datasets at the same time and doesn't have any assumption on the data structures, hence could deal with more complex cases than single affine transformation used in Procrustes analysis. (3) It could be easily extended to multi-manifolds alignment. In experiments, pose alignment on image sequences of different objects and face images, are used to verify the efficiency of our algorithm.

The rest of this paper is organized as follows. In Section 2, we first claim some notations and then we present our manifold alignment method in details. Section 3 shows the experimental results on two pose datasets of different objects and face images. Finally, Section 4 gives some concluding remarks.

2 Manifold Alignment via Corresponding Projections

2.1 Notations

Let us represent two datasets in matrix forms as $\mathbf{X} = [\mathbf{x}_1 \, \mathbf{x}_2 \cdots \mathbf{x}_M]$ and $\mathbf{Y} = [\mathbf{y}_1 \, \mathbf{y}_2 \cdots \mathbf{y}_N]$, where each column vector $\mathbf{x}_i \in \Re^{D_x}$ ($\mathbf{y}_j \in \Re^{D_y}$) denotes a data point in the input high-dimensional observation space. M and N indicate the numbers of the data points involved in the two data sets, respectively. The unknown shared low-dimensional embeddings of $\mathbf{X}$ and $\mathbf{Y}$ are denoted by $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_1 \, \tilde{\mathbf{x}}_2 \cdots \tilde{\mathbf{x}}_M]$ and $\tilde{\mathbf{Y}} = [\tilde{\mathbf{y}}_1 \, \tilde{\mathbf{y}}_2 \cdots \tilde{\mathbf{y}}_N]$, respectively. Suppose K labeled correspondence pairs are given in set C, *i.e.*, $(i,j) \in C$ if $\mathbf{x}_i$ corresponds to $\mathbf{y}_i$. With the prior knowledge, the original data sets can be separated into labeled and unlabeled subsets, denoted as $\mathbf{X} = [\mathbf{X}^l, \mathbf{X}^u]$ and $\mathbf{Y} = [\mathbf{Y}^l, \mathbf{Y}^u]$.

2.2 Overview Formulation

The target of our manifold alignment method is to learn the corresponding mappings which could project data points from different datasets to the intrinsic common embeddings. By doing this, we can compare the embeddings of the two datasets instead of their original high-dimensional representations.

More specifically, our algorithm learns mapping matrices $\mathbf{P}_x$ and $\mathbf{P}_y$ for data sets $\mathbf{X}$ and $\mathbf{Y}$, respectively. In mapping matrices learning, two important issues should be taken into consideration: (1) The common embeddings should be consistent with the given labeled correspondence pairs; (2) The common embeddings should preserve the local geometric structures in all original input spaces. We achieve the overall objective by minimizing the following energy function:

$$J(\mathbf{P}_x, \mathbf{P}_y) = J(\mathbf{P}_x, \mathbf{P}_y, \mathbf{X}^l, \mathbf{Y}^l) + \alpha_x J(\mathbf{P}_x, \mathbf{X}) + \alpha_y J(\mathbf{P}_y, \mathbf{Y}). \tag{1}$$

There are three terms in this energy function defined above. The first term $J(\mathbf{P}_x, \mathbf{P}_y, \mathbf{X}^l, \mathbf{Y}^l)$ is called the *correspondence preserving term*. And the other two terms, $J(\mathbf{P}_x, \mathbf{X})$ and $J(\mathbf{P}_y, \mathbf{Y})$, are the *manifold regularization terms* which are used to preserve the intrinsic manifold structures of different datasets. The parameters α_x and α_y are used to balance the three terms and further influence the relative contribution of local structure preserving in each observation space.

2.3 Energy Function Definition

In the objective function formulated in Eq.(1), the term of correspondence preserving cost is defined according to the given correspondence pairs as

$$J(\mathbf{P}_x, \mathbf{P}_y, \mathbf{X}^l, \mathbf{Y}^l) = \sum_{(i,j) \in C} ||\mathbf{P}_x^T \mathbf{x}_i - \mathbf{P}_y^T \mathbf{y}_j||^2. \tag{2}$$

As claimed in subsection 2.1, set C contains all index pairs (i,j) of given correspondences. Ideally, on the underlying common manifold, the embedded labeled pairs $(\mathbf{P}_x^T \mathbf{x}_i, \mathbf{P}_y^T \mathbf{y}_j)$ should be as close as possible. Therefore, we formulate the correspondence preserving term as a sum of squared differences (SSD) between the embeddings of all labeled correspondence data pairs.

Inspired by LLE [12] method, the two manifold regularization terms are defined as

$$J(\mathbf{P}_x, \mathbf{X}) = \sum_{i=1}^{M} ||\mathbf{P}_x^T \mathbf{x}_i - \sum_{k \in N^x(i)} w_{ik}^x \mathbf{P}_x^T \mathbf{x}_k||^2 \tag{3}$$

and

$$J(\mathbf{P}_y, \mathbf{Y}) = \sum_{j=1}^{N} ||\mathbf{P}_y^T \mathbf{y}_j - \sum_{k \in N^y(j)} w_{jk}^y \mathbf{P}_y^T \mathbf{y}_k||^2. \tag{4}$$

Here, w_{ik}^x is calculated through reconstructing the data point x_i using its k-nearest neighbors x_k ($k \in N^x(i)$) by minimizing $||\mathbf{x}_i - \sum_{k \in N^x(i)} w_{ik}^x \mathbf{x}_k||^2$ s.t. $\sum_{k \in N^x(i)} w_{ik}^x = 1$, and w_{jk}^y could be got in a similar manner. In the proposed method, we try to preserve the local topological structure of each input observation space.

Ultimately, we formulate the energy function as the following form:

$$\begin{aligned}
&J(\mathbf{P}_x, \mathbf{P}_y) \\
&= \sum_{(i,j) \in C} ||\mathbf{P}_x^T \mathbf{x}_i - \mathbf{P}_y^T \mathbf{y}_j||^2 + \alpha_x \sum_{i=1}^{M} ||\mathbf{P}_x^T \mathbf{x}_i - \sum_{k \in N^x(i)} w_{ik}^x \mathbf{P}_x^T \mathbf{x}_k||^2 \\
&\quad + \alpha_y \sum_{j=1}^{N} ||\mathbf{P}_y^T \mathbf{y}_j - \sum_{k \in N^y(j)} w_{jk}^y \mathbf{P}_y^T \mathbf{y}_k||^2.
\end{aligned} \tag{5}$$

The objective function gives a high penalty when the labeled correspondence pairs from different spaces or neighboring data points in the same space are mapped far apart. Therefore, it tries to find an underlying common manifold which reflects the intrinsic relationships and preserves the local geometric structures of data in the original spaces. In the following subsection, the solution of this optimization problem will be described.

2.4   Optimization Solution

Let us express the objective function Eq.(5) using a complete matrix form. More specifically, the correspondence preserving term defined in Eq.(2) is

$$J(\mathbf{P}_x, \mathbf{P}_y, \mathbf{X}^l, \mathbf{Y}^l) = \sum_i^M \sum_j^N ||\mathbf{P}_x^T \mathbf{x}_i - \mathbf{P}_y^T \mathbf{y}_j||^2 \mathbf{L}_{ij}^{xy}. \tag{6}$$

Here the matrix $\mathbf{L}^{xy}$ with size of $M \times N$ contains the prior information on labeled correspondence pairs. The correspondence indicator matrix is defined as

$$\mathbf{L}_{ij}^{xy} = \begin{cases} 1 & (i,j) \in C \\ 0 & otherwise \end{cases}. \tag{7}$$

Then, we can have the matrix form of Eq.(6) as follows:

$$J(\mathbf{P}_x, \mathbf{P}_y, \mathbf{X}^l, \mathbf{Y}^l) = \text{Tr}\left(\mathbf{P}_x^T \mathbf{X} \mathbf{L}^x \mathbf{X}^T \mathbf{P}_x + \mathbf{P}_y^T \mathbf{Y} \mathbf{L}^y \mathbf{Y}^T \mathbf{P}_y\right) - \text{Tr}\left(\mathbf{P}_x^T \mathbf{X} \mathbf{L}^{xy} \mathbf{Y}^T \mathbf{P}_y + \mathbf{P}_y^T \mathbf{Y} (\mathbf{L}^{xy})^T \mathbf{X}^T \mathbf{P}_x\right). \tag{8}$$

Here, $\text{Tr}(\cdot)$ is the trace operator. The matrices $\mathbf{L}^x$ and $\mathbf{L}^y$ are defined based on the matrix $\mathbf{L}^{xy}$ as

$$\mathbf{L}_{ii}^x = \sum_j \mathbf{L}_{ij}^{xy} \quad \text{and} \quad \mathbf{L}_{jj}^y = \sum_i \mathbf{L}_{ij}^{xy}, \tag{9}$$

and the off diagonal entries in the two matrices are all zero.

As for the two manifold preserving terms, we have similar matrix forms as

$$J(\mathbf{P}_x, \mathbf{X}) = \sum_{i=1}^{M} ||\mathbf{P}_x^T \mathbf{x}_i - \sum_j w_{ij}^x \mathbf{P}_x^T \mathbf{x}_j||^2 = \mathrm{Tr}\left(\mathbf{P}_x^T \mathbf{X} \mathbf{M}^x \mathbf{X}^T \mathbf{P}_x\right) \tag{10}$$

and

$$J(\mathbf{P}_y, \mathbf{Y}) = \sum_{i=1}^{N} ||\mathbf{P}_y^T \mathbf{y}_i - \sum_j w_{ij}^y \mathbf{P}_y^T \mathbf{y}_j||^2 = \mathrm{Tr}\left(\mathbf{P}_y^T \mathbf{Y} \mathbf{M}^y \mathbf{Y}^T \mathbf{P}_y\right). \tag{11}$$

The matrix $\mathbf{M}^x$ with size of $M \times M$ and $\mathbf{M}^y$ with size of $N \times N$ serve to preserve the local linear structures between data points in $\mathbf{X}$ and $\mathbf{Y}$, respectively. Similar representations are used in LLE [12] and NPE [8]. As for $\mathbf{M}^x$, it could be presented as

$$\mathbf{M}^x = (\mathbf{I} - \mathbf{W}^x)^T (\mathbf{I} - \mathbf{W}^x). \tag{12}$$

In Eq.(12), the reconstruction weights matrix $\mathbf{W}^x$ is defined as

$$\mathbf{W}_{ij}^x = \left\{ \begin{array}{ll} w_{ij}^x & j \in N^x(i) \\ 0 & otherwise \end{array} \right. , \tag{13}$$

and $j \in N^x(i)$ means that x_j belongs to x_i's k-nearest neighbors. The matrix $\mathbf{M}^y$ is defined in a similar way on $\mathbf{Y}$.

Combining the new forms of all objective terms, the energy function is rewritten as

$$\begin{aligned} &J(\mathbf{P}_x, \mathbf{P}_y) \\ &= J(\mathbf{P}_x, \mathbf{P}_y, \mathbf{X}^l, \mathbf{Y}^l) + \alpha_x J(\mathbf{P}_x, \mathbf{X}) + \alpha_y J(\mathbf{P}_y, \mathbf{Y}) \\ &= \mathrm{Tr}\left([\ \mathbf{P}_x^T \mathbf{X} \quad \mathbf{P}_y^T \mathbf{Y}\ ] \left[\begin{array}{cc} \mathbf{L}^x + \alpha_x \mathbf{M}^x & -\mathbf{L}^{xy} \\ -(\mathbf{L}^{xy})^T & \mathbf{L}^y + \alpha_y \mathbf{M}^y \end{array} \right] \left[\begin{array}{c} \mathbf{X}^T \mathbf{P}_x \\ \mathbf{Y}^T \mathbf{P}_y \end{array} \right] \right). \end{aligned} \tag{14}$$

Let $\mathbf{P} = \left[\begin{array}{c} \mathbf{P}_x \\ \mathbf{P}_y \end{array} \right]$, $\mathbf{Z} = \left[\begin{array}{cc} \mathbf{X} & \\ & \mathbf{Y} \end{array} \right]$ and $\mathbf{A} = \left[\begin{array}{cc} \mathbf{L}^x + \alpha_x \mathbf{M}^x & -\mathbf{L}^{xy} \\ -(\mathbf{L}^{xy})^T & \mathbf{L}^y + \alpha_y \mathbf{M}^y \end{array} \right]$. Consequently, we obtain a concise form as

$$J(\mathbf{P}_x, \mathbf{P}_y) = \mathrm{Tr}\left(\mathbf{P}^T \mathbf{Z} \mathbf{A} \mathbf{Z}^T \mathbf{P}\right). \tag{15}$$

Finally, we add the constraints to achieve scaling and translation invariance, and solve the optimization problem of manifold alignment by minimizing

$$J(\mathbf{P}_x, \mathbf{P}_y) \ \text{s.t.} \ \frac{1}{M+N} \mathbf{P}^T \mathbf{Z} \mathbf{Z}^T \mathbf{P} = \mathbf{I} \ \text{and} \ \mathbf{P}^T \mathbf{Z} \mathbf{e} = 0, \tag{16}$$

where $\mathbf{I}$ is the identity matrix with size of $d \times d$, and $\mathbf{e} = [1\ 1\ \cdots\ 1]^T$ is the vector of ones with $M + N$ entries.

Let $\mathbf{E} = \mathbf{Z} \mathbf{A} \mathbf{Z}^T$ and $\mathbf{F} = \mathbf{Z} \mathbf{Z}^T$. The solution to the optimization problem with respect to $\mathbf{P}$ is given by the second to $(d+1)$-th smallest generalized eigenvectors $\mathbf{p}$ with $\mathbf{E}\mathbf{p} = \lambda \mathbf{F}\mathbf{p}$. Note that $\mathbf{p} = \left[\begin{array}{c} \mathbf{p}_x \\ \mathbf{p}_y \end{array} \right]$ and d represents the dimensionality of the common embeddings.

The final mapping matrices $\mathbf{P}_x$ and $\mathbf{P}_y$ are composed of the corresponding d eigenvectors, $\mathbf{p}$'s. More specifically, $\mathbf{P}_x$ with size of $D_x \times d$ is constructed by arraying the $\mathbf{p}_x$'s as its d column vectors and $\mathbf{P}_y$ with size of $D_y \times d$ is constructed by arraying the $\mathbf{p}_y$'s as its d column vectors.

During the optimization process, the following issue should be concerned. In general case, $\mathbf{F} = \mathbf{ZZ}^T$ is singular. There are two ways to assure $\mathbf{F}$ invertible. One is to incorporate a regularization term such as $\mathbf{F} = \mathbf{ZZ}^T + \kappa\mathbf{I}$, where κ is set to a small positive value (e.g., $\kappa = 10^{-6}$). The other is to carry out dimensionality reduction on $\mathbf{X}$ and $\mathbf{Y}$ respectively, such as PCA, to reduce the singularity of matrix $\mathbf{F}$.

2.5 The algorithm

The algorithm of manifold alignment via correspondence projection is summarized in Table 1. In our method, two separate mapping matrices $\mathbf{P}_x$ and $\mathbf{P}_y$ for different datasets are computed at the same time by solving a convex optimization problem in Step 1. Then the learned explicit projections can be easily generalized to new test data in Step 2. Therefore, our method is very effective by learning mappings from each input space to the shared low-dimensional space explicitly and simultaneously. Furthermore, it is suitable for real-time systems due to the efficiency.

Input: Datasets $\mathbf{X} = [\mathbf{X}^l, \mathbf{X}^u]$, $\mathbf{Y} = [\mathbf{Y}^l, \mathbf{Y}^u]$, given correspondence pairs in C,
   and query data point $\mathbf{y}$.
Output: $\mathbf{y}$'s counterpart in $\mathbf{X}$.
  Step 1: Learning mapping matrices $\mathbf{P}_x$ and $\mathbf{P}_y$ according to Eq. (16);
  Step 2: Compute the embedding for $\mathbf{y}$ by $\tilde{\mathbf{y}} = \mathbf{P}_y^T \mathbf{y}$;
    Compute the embeddings for $\mathbf{X}$ by $\tilde{\mathbf{X}} = \mathbf{P}_x^T \mathbf{X}$;
  Step 3: Find $\tilde{\mathbf{y}}$'s the nearest neighbor $\tilde{\mathbf{x}}_i$ from the column vectors of $\tilde{\mathbf{X}}$;
  Step 4: $\mathbf{x}_i$ corresponding to $\tilde{\mathbf{x}}_i$ is treated as the counterpart for $\mathbf{y}$.

Table 1: Algorithm of manifold alignment via correspondence

3 Experiments

In this Section, we verify the efficiency of our manifold alignment method on multi-pose sequences alignment problem. In our experiments, we attempt to align the images with various poses of different objects and persons.

3.1 Experimental Settings

For comprehensive comparison, the proposed manifold alignment algorithm is compared with some state-of-the-art methods on two multi-pose datasets: COIL-20 [2] and FACE-10.

As illustrated in Figure 1(a), the first pose dataset, COIL-20, contains 1440 images of 20 different objects. The pose coordinates for each object specify the camera movements around it, which are at interval of $5°$ and contain 72 different sites in total, as shown in Figure 1(c). The examples of the first object are shown in Figure 1(b). In our experiment, 32 images for each observation are selected evenly for the training set and the rest 40 images for testing. The second pose dataset, FACE-10, contains face image sequences with various poses (from $-90°$ to $+90°$ as indicated in Figure 1(e)) of 10 persons. Two example sequences are illustrated in Figure 1(d). In our experiment, 37 images for each observation are selected evenly for the training set and the rest images for testing. We use the image intensity as the feature. The images in COIL-20 and FACE-10 are of size 16×16 and 32×32 pixels, respectively.
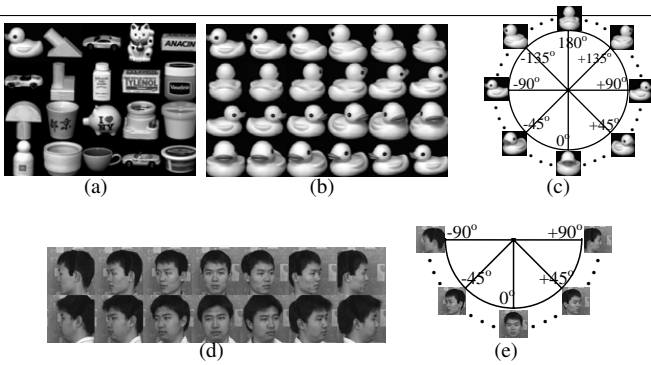
Figure 1: (a) Sample images for 20 subjects in COIL-20 dataset; (b) one example sequence in COIL-20 dataset; (c) coordinates of the poses in COIL-20 dataset; (d) two example sequences of 'person_01' and 'person_02' in FACE-10;(e) coordinates of the poses in FACE-10 dataset.

We compare our proposed algorithm with some baseline and previous related methods. More specifically, five approaches are included in our comparative study: (1) Baseline method which directly uses the original space for alignment; (2) Locality Preserving Projections (LPP) [7]; (3) Nonlinear Manifold Alignment(NMA) [10]; (4) Linear Procrustes Analysis (LPA) [16] ; (5) Our approach. For the last four methods mentioned above, after finding the shared embedding space, a simple nearest neighbor (NN) classifier is performed and the nearest pairs are aligned as the counterparts. In the next subsection, both qualitative and quantity are conducted. Due to space limitation, the Euclidean and LPP results are omitted for qualitative comparisons. While in quantity analysis, we will give comparative studies for all mentioned methods.

There are a few parameters involved in our experiments. As for the number of neighbors k used to calculate the weights in Eq.(5), we empirically set $k = 2$ for COIL-20 and $k = 6$ for FACE-10, respectively. The dimensionality of the shared common embeddings is fixed to 5. And the balance coefficients α_x and α_y are set to some values in $(0,1]$ which can be finally determined by cross-validation. In the comparative studies, the common parameters of other methods are set the same as ours.

3.2 Pose Alignment Results and Analysis

For clearer comparison of the alignment results, we inlay two aligned sequences on two concentric circles. And then, we connect the points in the sequence $\mathbf{Y}$ with their corresponding points in the sequence $\mathbf{X}$. As shown in Figure 2, the 'blue bold' lines connect the labeled point pairs. The 'red' lines denote the connections for aligned unlabeled points. In perfect alignment, the connections should be along the direction of radius (of the rays emitting from the center of the concentric circles). Under this visual criterion, our method get the comparable and even better performance compared with the semi-supervised algorithm NMA. Meanwhile, LPA cannot give satisfactory results for this situation. As illustrated in Figure 2(b), even if there are only 4 labeled pairs, our method still achieves good results. The more pairs given, the better performance our method achieves. Some results of face image sequences with various poses are shown in Figure 3.

One advantage of our algorithm is that it could deal with new test data in a direct way as mentioned in subsection 2.5. In Figure 4, the 'blue bold' lines connect the labeled point pairs; the 'black dashed' lines denote the connections for the aligned unlabeled data in training set;
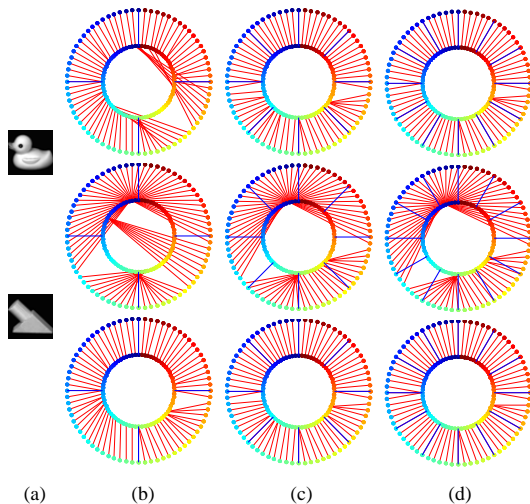
Figure 2:   The results of aligning 'duck' and 'block' inlayed on two concentric circles. (a) Examples in two sequences; The rows from top to bottom correspond the results of the algorithms NMA [10], LPA [16], and our method, respectively. For each column (b), (c) and (d), the number of the labeled correspondence pairs is 4, 8 and 12, respectively.
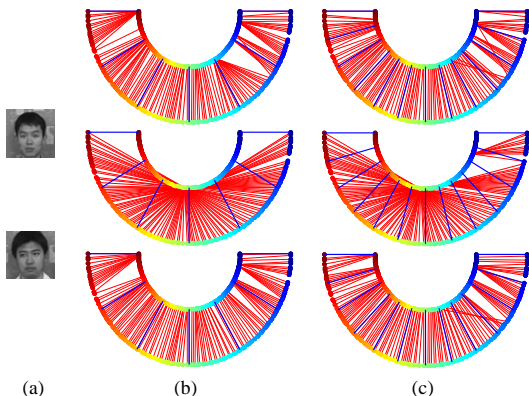

Figure 3:   The results of aligning 'person_01' (138 images) and 'person_02' (134 images) inlayed on two concentric circles. (a) Examples in two sequences; The rows from top to bottom correspond the results of the algorithms in NMA [10], LPA [16], and our method, respectively. For each column (b) and (c), the number of the labeled correspondence pairs is 7 and 13, respectively.

and the 'red' lines denote the connections for the aligned new test data. From Figure 4, we can see that both results of our method for the unlabeled training data and the new test data are more accurate and stable than the algorithm LPA.
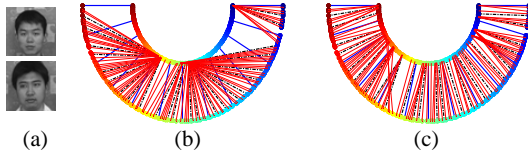

(a)　　　　　　(b)　　　　　　　　(c)

Figure 4: The results for handling new test data. (a) Examples in two sequences; (b) the results of method LPA [16]; (c) our results.

Some quantity results on both databases are summarized in Table 2 and Table 3. In the experiments, we use the first sequence ('duck', 'person_01') as **X** and other sequences as **Y**. For each configuration with different label correspondence pairs, we report the average pose estimation errors and standard derivations. As shown in these two tables, the baseline method, which doesn't consider the relationship between the two manifolds, yields poor performance. The classical LPP method, which can uncover the essential manifold structure in a single observation space, fails to well handle the scenario of two (or multiple) observation spaces. The NMA method achieve relative good results, but it cannot process the new test data since no explicit mapping is learned. Meanwhile, LPA still cannot give satisfactory results in these situations since the relationship between two embedded manifolds is beyond affine transformation. Among all methods mentioned above, the proposed method achieves the lowest errors. And the error rates decreases with the increasing number of the labeled correspondence pairs.

Method	$\sharp$ labels = 8		$\sharp$ labels = 16	
	Unlabeled Error	Test Error	Unlabeled Error	Test Error
Baseline	36.897±3.516	42.237±3.770	36.694±4.546	42.237±3.770
LPP	34.540±5.74	39.855±5.34	34.540±5.74	39.855±5.34
NMA	18.43±11.202	-	12.65±12.766	-
LPA	18.706±9.787	17.625±8.32	17.747±8.998	17.434±8.707
Ours	**15.57±12.079**	**14.533±9.147**	**10.674±10.443**	**9.934±7.834**

Table 2: The pose estimation errors (unit: degree) on COIL-20 data set (mean±std-dev)

Method	$\sharp$ labels = 7		$\sharp$ labels = 13	
	Unlabeled Error	Test Error	Unlabeled Error	Test Error
Baseline	62.326±7.516	68.945±15.409	62.079±7.912	68.945±15.409
LPP	37.616±19.496	39.344±16.925	37.616±19.496	39.344±16.925
NMA	9.089±0.411	-	3.366±0.228	-
LPA	32.348±15.277	28.555±13.805	32.273±17.245	28.549±15.431
Ours	**6.933±0.585**	**14.438± 4.313**	**2.630±0.544**	**9.032±5.065**

Table 3: The pose estimation errors (unit: degree) on FACE-10 face data set (mean±std-dev)

3.3   Extensions

So far, all experiments performed in this paper are limited to the alignment of two manifolds. Actually, our method could be extended to multi-manifold alignment according to the formulation in Section 2. In Figure 5, we show an illustrative example result of three-manifold alignment. Note that, LPA method cannot possess this natural extension.
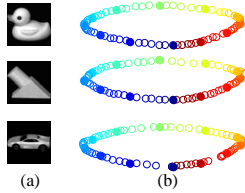

(a)   (b)

Figure 5: The results of alignment for three datasets. (a) Examples in three sequences; (b) our alignment results.

4   Concluding Remarks

This paper proposes a novel manifold alignment method via corresponding projections. Different from most previous work, our method learns the explicit corresponding projections from each observation space to the common embedding space. Benefiting from the explicit projections, our method could handle new test data directly rather than re-aligning the manifolds. Compared with the method based on some assumptions about the structure of dataset (e.g., alignment using Procrustes analysis), our method could deal with more complex real-world alignment problem. Besides, our method could be easily extended to align multi-manifolds. Experimental results on pose datasets of various objects and face images show that our method gets comparable and even better performance than other related alignment methods.

5   Acknowledgement

References

[1] R. A. Horn and C. R. Johnson. Matrix Analysis. Cambridge University Press, Cambridge, 1990.

[2] S. A. Nene, S. K. Nayar, H. Murase. Columbia Object Image Library (COIL-20), Technical Report CUCS-005-96, February 1996.

[3] Y. Bengio, J. Paiement and P. Vincent. Out-of-Sample Extensions for LLE, Isomap, MDS, Eigenmaps, and Spectral Clustering. In Advances in Neural Information Processing Systems, pp 177-184, 2004.

[4] M. Belkin and P. Niyogi. Laplacian eigenmap and spectral techniques for embedding and clustering. In Advances in Neural Information Processing Systems, volume 14, 2002.

[5] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15:1373-1396, 2003.

[6] Haifeng Gong, Chunhong Pan, Qing Yang, Hanqing Lu, and Songde Ma. A Semi-Supervised Framework for Mapping Data to the Intrinsic Manifold. Proceedings of the Tenth IEEE International Conference on Computer Vision, pp 98 - 105, 2005.

[7] Xiaofei He and Partha Niyogi. Locality preserving projections. In Advances of Neural Information Processing Systems (NIPS) 16, 2003.

[8] Xiaofei He and Deng Cai and Shuicheng Yan and Hong-Jiang Zhang. Neighborhood preserving embedding. Proceedings of the Tenth IEEE International Conference on Computer Vision, pp 1208-1213, 2005.

[9] J. Ham, D. D. Lee, and L. K. Saul. Learning high-dimensional correspondences from low-dimensional manifolds. In Workshop on The Continuum from Labeled to Unlabled Data in Machine Learning and Data Mining at 20-th International Conference on Machine Learning, pp 34-39, 2003.

[10] J. Ham, D. D. Lee, and L. K. Saul. Semisupervised alignment of manifolds. Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics, pp. 120-127, 2005.

[11] Aaron P. Shon, Keith Grochow, Aaron Hertzmann, and Rajesh P. N. Rao. Learning shared latent structure for image synthesis and robotic imitation. In Advances of Neural Information Processing Systems, pp 1233-1240, 2006.

[12] S. Roweis and L. Saul. Nonlinear dimensionality reduction by locally linear embedding. Science, 290:2323-2326, 2000.

[13] A. S. Georghiades, P. N. Belhumeur, D. J.and Kriegman. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose. IEEE Trans. Pattern Anal. Mach. Intelligence, 23(6): 643-660, 2001.

[14] J. S. Taylor and N. Cristianini. Kernel Methods for Pattern Analysis. Cambridge University Press, 2004.

[15] J. Tenenbaum, V. de Silva, and J. Langford. A global geometric framework for nonlinear dimensionality reduction. Science, 290:2319-2323, 2000.

[16] Chang Wang, Sridhar Mahadevan. Manifold alignment using Procrustes analysis. Proceedings of the 25th international conference on Machine learning, pp 1120-1127, 2008.

[17] Liang Xiong, Fei Wang and Changshui Zhang. Semi-definite Manifold Alignment. Proceedings of the 18th European conference on Machine Learning, pp 773 - 781, 2007.

[18] J.J.Verbeek. Learning nonlinear image manifolds by global alignment of local linear models. IEEE Trans. Pattern Analysis and Machine Intell., 28(8):1236-1250,2006.