Neurocomputing ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Local subspace smoothness alignment for constrained local model fitting

Dakun Liu [a,b], Xiaoyang Tan [a,b,*]

[a] College of Computer Science & Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, PR China
[b] Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 210016, PR China

ARTICLE INFO

Article history:
Received 2 February 2016
Received in revised form
23 June 2016
Accepted 5 July 2016
Communicated by Dacheng Tao

Keywords:
Facial landmark estimation
Constrained local models
Manifold learning
Sparse representation

ABSTRACT

Constrained local model (CLM) is a classic method for facial landmarks estimation. While the CLM enhances the well-known active shape model with discriminative local appearance models, its shape model is based on the point distribution model, which is essentially principal component analysis over the training facial shape vectors and hence the nonlinear manifold of facial shapes is not well embedded. In this paper, we propose a novel manifold learning method, i.e., local subspace smoothness alignment (LSSA), to address this issue. The LSSA approach smoothes the nonlinear structure directly in the original feature space, with a newly defined geometric measure for the curvature of the local structures. We then proceed to apply this method for face alignment, with an ensemble of correlated local subspaces derived from LSSA. The proposed method is demonstrated on both toy data and real-world datasets that it yields reasonable manifold embedding and leads to encouraging performance for face alignment even under difficult conditions.

1. Introduction

In face recognition, the representation of face images is a very important issue to be addressed. Although the gray-scale face images can be directly used as input in some methods such as non-negative matrix factorization [1], sparse representation classification [2], etc., they usually assume that these face images are cropped or the facial features in different images with the same semantics are well aligned. However, this is usually not the case in practice. In fact, the problem of aligning facial feature points is so difficult that it is a separate research topic in the field of face recognition [3–8], called face alignment or facial landmarks estimation. Nowadays the successful registration and tracking of non-rigidly varying geometric landmarks on face has become a key ingredient to an automatic facial analysis system [9–12].

The challenges of facial landmarks estimation mainly come from the variety of appearance patches centered on the landmarks, such as lighting, occlusion, expression and so on. Many approaches for accurate non-rigid facial registration and face tracking focus on building a synthesis model to reconstruct the landmarks of a possibly unseen face image based on the facial shapes and appearances of training images. One of the most famous models is the active shape model (ASM) [13], which derives the positions of landmarks based on the statistical information of landmarks distribution. In ASM, the point distribution model (PDM) [13] is used to model the valid shape space of face landmarks with a set of deformation parameters.

Constrained local model (CLM) [14] is another famous approach for non-rigid face registration/tracking. CLM is a generalization of ASM in the sense that the searching space of ASM for potential facial landmarks is 1D, while the CLMs are based on the 2D response map. 2D response maps are usually estimated by a discriminative local appearance model and can better capture appearance information around facial landmarks, and this information, if used wisely, should give better results.

Many CLM variations have been proposed recently. These methods pursue the same goal as CLM but use more robust and complex models based on the distribution of landmarks response. Particularly, the searching strategy of the original CLM is based on the hypothesis that the locations of facial landmarks obey a distribution of isotropic Gaussian, which is obviously not so realistic. So many methods consider anisotropic Gaussian instead [15–17]. Although this anisotropic Gaussian approximation of the response maps effectively overcomes some drawbacks of its isotropic counterpart, sometimes their performance can be poor especially when the facial appearance changes a lot. To address it, some other models are investigated as well, such as Gaussian mixture model (GMM) [18] and nonparametric model [19]. Additionally, some works focus on improving the quality of response maps [20–

* Corresponding author at: College of Computer Science & Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, PR China.
E-mail address: x.tan@nuaa.edu.cn (X. Tan).

http://dx.doi.org/10.1016/j.neucom.2016.07.007
0925-2312/

23]. These methods have a common characteristic, that is dividing and conquering, independently training a special local model (detector, regressor, or part template) for each feature point. So they are known as local methods.

Relatively, the methods which consider all the feature points as a whole, rather than treat them as conditionally independent are regarded as holistic method. Active appearance models (AAMs) [24] is the representative method. It simultaneously models the intrinsic variation in both appearance and shape as a linear combination of basis models of variation. Among the holistic methods, explicit shape regression (ESR) [25], supervised descent method (SDM) [26], ensemble of regression trees (ERT) [27] and local binary feature (LBF) [3] are four state-of-the-art methods. All of them performed under the cascaded shape regression framework using shape-indexed features. ESR directly learns a regression function to infer the shape from a sparse subset of pixel intensities indexed relative to current shape estimate, while ERT substitutes the weak fern regressor in ESR with a 4 regression tree which further improves the performance. SDM employs a cascaded linear regression to estimate the shape based on hand-designed SIFT feature, while LBF learns a set of highly discriminative local binary features for each feature point independently, and then uses the learned features jointly to learn a linear regression for the final prediction, which is highly efficient and achieves very accurate performance.

Despite these methods archived partial successes in face alignment, the limitation of CLMs still remains. Particularly, most CLM based models use the PDM to model the shape space. The PDM is essentially a linear approximation to the shape of a non-rigid object deformations with a global rigid transformation. Compared with the complexity of the design on the response distribution, the PDM model is too rough – actually, due to the highly nonlinear and non-convex of the facial shape space, linear analysis used by PDM is far from adequate.

In this paper, we propose a novel manifold learning method, i.e., local subspace smoothness alignment (LSSA), to address this issue. The LSSA approach smoothes the nonlinear structure directly in the original feature space, with a newly defined geometric measure for the curvature of the local structures. After performing the LSSA transformation, we use the adjacent shapes for CLM fitting in ensemble of correlated local subspaces.

This paper is organized as follows: the background on PDM and manifold learning are described in Section 2. The motivation and details of local subspace smoothness alignment are described in Section 3. CLM fitting with an ensemble of local subspaces learnt from LSSA is given in Section 4. Comparison experiments on the works of manifold learning and extensive experiments on demonstrating the importance of the prior on manifold in CLM fitting are shown in Section 5; we conclude this paper in Section 6 at last.

2. Background

2.1. The point distribution model

Both ASM and CLM use the point distribution model (PDM) to model the shape space. Specifically, based on the principal component analysis method (PCA), the PDM reconstructs the facial shape of an unseen face image linearly:

$$\mathbf{x} = s\mathbf{R}(\bar{\mathbf{x}} + \mathbf{\Phi q}) + \mathbf{t}, \tag{1}$$

where $\mathbf{R}$, s and $\mathbf{t}$ control the rigid rotation, scale and translations respectively while $\mathbf{q}$ controls the non-rigid variations of the shape and $\mathbf{\Phi}$ denotes the submatrix of the basis of variations. Then all the parameters of the shape model can be denoted as $\mathbf{p} = \{s, \mathbf{R}, \mathbf{t}, \mathbf{q}\}$, where the rigid transformation parameter $\mathbf{q}$ is often assumed to exhibit a Gaussian distribution while the non-rigid transformation parameters s, $\mathbf{R}$ and $\mathbf{t}$ that place the model in the image are all assumed uniform distributions. The goal of this PCA-based CLM is to reconstruct the face shape $\mathbf{x}$ from the parameters $\mathbf{p}$.

This PDM is convenient but oversimplifies the distribution of facial shapes. To visualize the distribution of facial shapes, we apply the ISOMAP [28] method on a set of facial shapes from the PUT dataset. The PUT database contains 9971 images from 100 subjects, and each image is annotated with a 194-point markup as ground truth landmarks. Here we choose 63 landmarks in each image and 1200 random facial shapes. Fig. 1 gives the learnt manifold, where each point denotes one facial shape, and the corresponding facial shapes of the red ones are illustrated along the coordinate axes. It can be seen that along the X-axis the facial orientation changes from left to right, while along the positive direction of the Y-axis, the facial shapes become "thinner".

2.2. Manifold learning

The objective of manifold learning is to model the true geometric distribution of data. Although these algorithms could all be categorized into graph embedding framework [29], they are rarely regarded as a whole due to their different motivations and efficiency. According to the directions on the research of manifold learning, they are mainly of two kinds. The first one can be called the "direct" approach, i.e., distance preservation. These approaches focus on learning a low-dimensional space and keeping the similarity relation between data. Typical methods include MDS (*multidimensional scaling* [30]) and ISOMAP (*isometric mapping* [28]). The second one is an "indirect" approach, i.e., linear locality preservation or patch alignment [31]. Based on the hypothesis that the local structure of data is linear, these approaches try to preserve the locality. Typical methods along this line include LLE (*locally linear embedding* [32]), LPP (*Locality preserving projections* [33]), LTSA (*local tangent space alignment* [34]), and so on.

Compared with the distance-preserving methods, linear-locality preservation is more simple and effective in practice. One of the most popular algorithms is the LPP [33]. It is considerably fast and has an explicit projection function, which, however, is a global linear projection and hence is too rigid to handle the non-linearity of data. By contrast, the LTSA [34] constructs the local coordinates from each local patch and learns a set of linear projections for them. After that, an alignment operation is used to combine the local coordinates into a global one.

However, there are still many difficulties when applying these manifold methods in the real-world applications. Besides the challenges concerning the generalization, such as dealing with sparse, non-uniform data or incorporating discriminant information into the model [35–39], finding the number of inherent dimensionality is another important problem that needs to be addressed, which is unfortunately still open to now. Although some ad hoc methods can be used sometimes, if the dimensionality of data is considerably high, it is quite time-consuming to estimate the dimensionality of manifold. Meanwhile, it is inevitable to loss some useful information during dimension reduction.

3. Local subspace smoothness alignment (LSSA)

In this section, we describe our local subspace smoothness alignment method, which overcomes some limitations of the traditional manifold learning methods.

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007

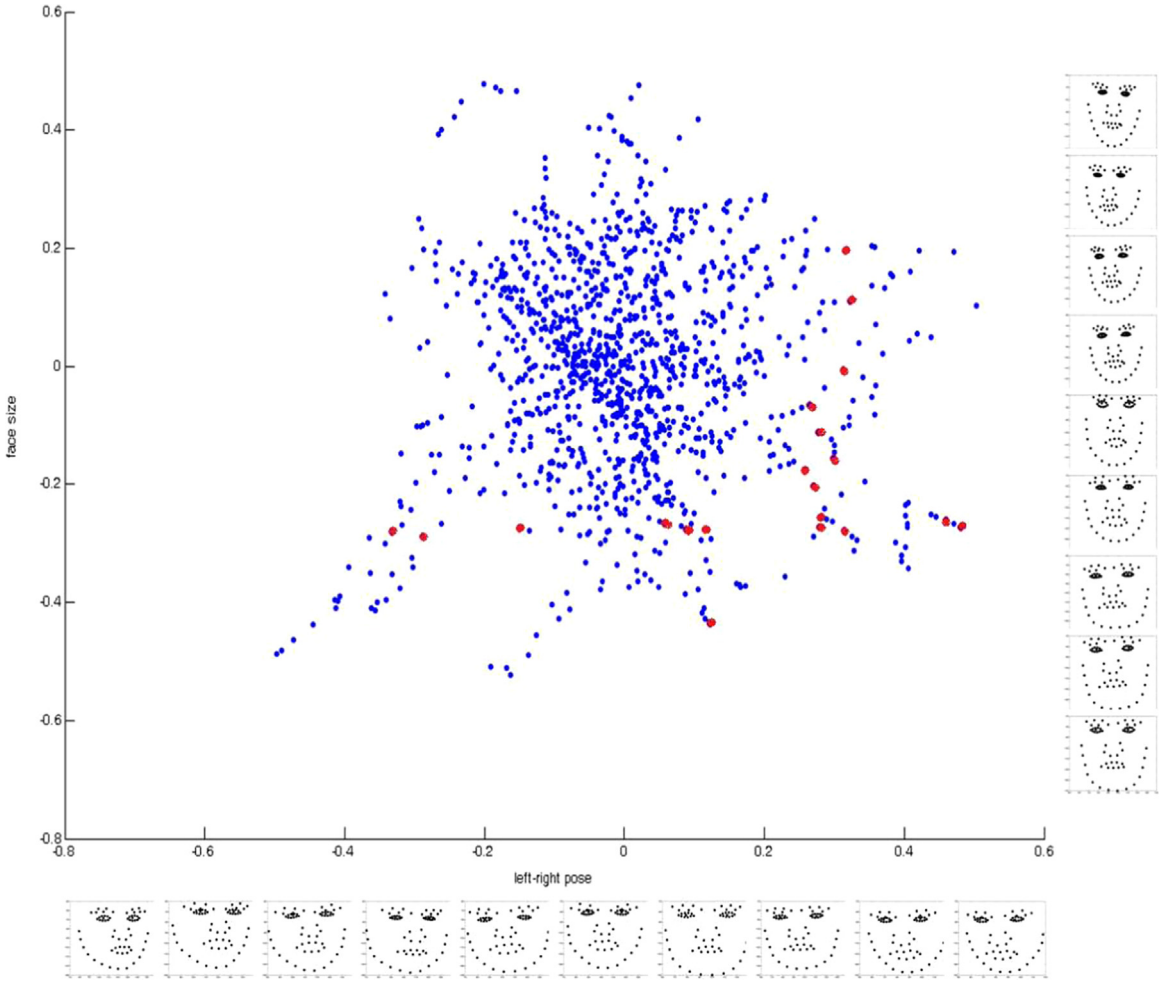


Fig. 1. Visualization of the distribution of facial shapes in 2D space. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

3.1. The motivation

The method is inspired by the local tangent space alignment (LTSA) [34], but the key idea is to directly "unfold" the non-linear data structure in the original feature space. For this purpose, we first propose a method to measure the geometric structure of data based on the following observations. That is, for each local subspace or the neighborhood, when the sum of the differences between the midpoint and other points in the same neighborhood is small, the curvature of the local structure is relatively smooth. Otherwise the curvature of the local structure is large.

Fig. 2 demonstrates the idea in two-dimensional space, where

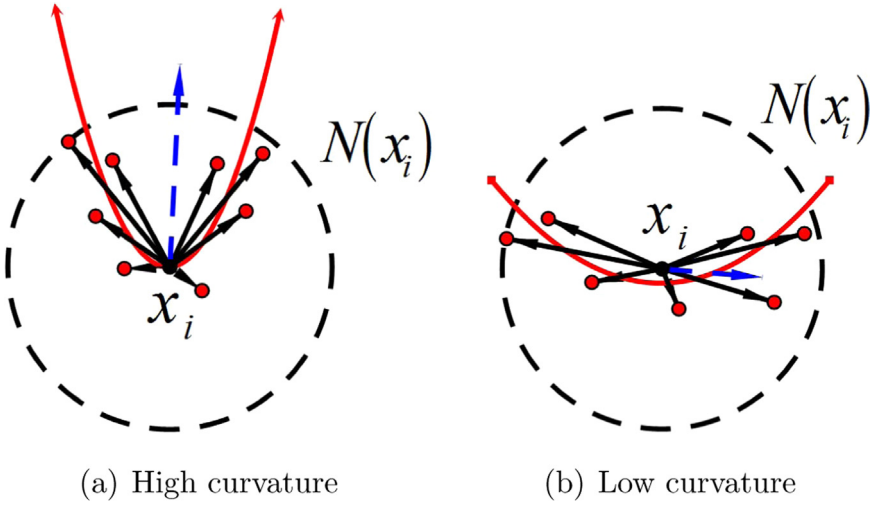


Fig. 2. Conceptual illustration of the local subspace smoothness alignment in 2D space. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007

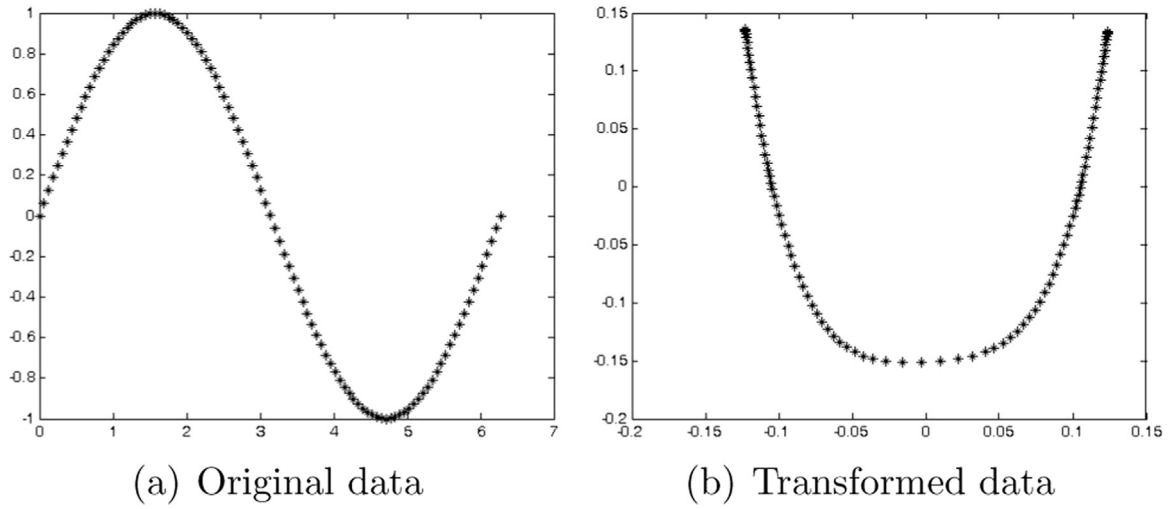


Fig. 3. Illustration of local subspace smoothness alignment on an artificial dataset.

the black point $\boldsymbol{x}_i$ denotes the midpoint of interest, and the red points denote its neighbors inside some neighborhood $N(\boldsymbol{x}_i)$ which is marked by the dashed circle. Assume that the real local geometric structure of the data is represented by the red curve, and black arrows respectively denote the difference vectors between the midpoint and all the remaining points in the neighborhood, while the net vector (i.e., the sum of these difference vectors) is denoted by the blue dash arrow. One can see that, the norm of the net vector in the left figure is much larger than that in the right figure, indicating that the curvature of the former figure is higher than that of the latter one.

3.2. The method

Based on the above observations, to smooth the local structures, one just needs to constrain the norm of the net vector of each neighborhood. Hence the model of local structure smoothness is formulated as follows:

$$\min_{\boldsymbol{\tau}_{ij},\boldsymbol{L}_c} \sum_i \left(\sum_j \| \boldsymbol{\tau}_{ij} - \boldsymbol{L}_c^{(i)} \boldsymbol{x}_{ij} \|^2 + \left\| \sum_j (\boldsymbol{\tau}_{ij} - \boldsymbol{\tau}_i) \right\|^2 \right) \tag{2}$$

In the above objective, we use i to denote the index of the local structure, and j denotes the index of the point in the local structure. Now $\boldsymbol{x}_i$ is the midpoint of the structure in the feature space, while $\boldsymbol{x}_{ij}$ is one of the point in the neighborhood of $\boldsymbol{x}_i$. We assume that $\boldsymbol{x}_{ij} \in N(\boldsymbol{x}_i)$, but $\boldsymbol{x}_{ij} \neq \boldsymbol{x}_i$. In addition, we use $\boldsymbol{\tau}_{ij}$ and $\boldsymbol{\tau}_i$ to denote the transformed points corresponding to $\boldsymbol{x}_{ij}$ and $\boldsymbol{x}_i$ respectively and $\boldsymbol{L}_c^{(i)}$ is the local affine transformation of the local structure. Our goal is to learn the transformation matrix $\boldsymbol{L}_c^{(i)}$ and the locations $\{\boldsymbol{\tau}_{ij}\}$ of the transformed points from the data.

Note that the first term of Eq. (2) represents the relationship between the original local structure and transformed one. That is, the smoothed points are derived by local affine transformation from the original points. The second term is the smoothness constraint. For the efficiency of optimization, we first get the upper bound of the smoothness term:

$$\begin{aligned} \left\| \sum_j (\boldsymbol{\tau}_{ij} - \boldsymbol{\tau}_i) \right\|^2 &\leq \sum_j \| \boldsymbol{\tau}_{ij} - \boldsymbol{\tau}_i \|^2 \\ &= \mathrm{trace}[(\boldsymbol{T}_i - \boldsymbol{L}_c^{(i)} \boldsymbol{x}_i \boldsymbol{e}_k^T)^T (\boldsymbol{T}_i - \boldsymbol{L}_c^{(i)} \boldsymbol{x}_i \boldsymbol{e}_k^T)] \\ &= \| \boldsymbol{T}_i - \boldsymbol{L}_c^{(i)} (\boldsymbol{x}_i \boldsymbol{e}_k^T) \|_F^2 \end{aligned} \tag{3}$$

where $\boldsymbol{\tau}_i = \boldsymbol{L}_c^{(i)} \boldsymbol{x}_i$, $\boldsymbol{T}_i$ is the transformed data matrix of the local structure with each column a datum, $\boldsymbol{e}_k$ is an unit column vector with the length k, and k is the number of points in the neighborhood. Now let $\boldsymbol{X}_i$ denote the data matrix of the local structure in the original feature space, the objective equation (2) is rewritten as follows:

$$\min_{\boldsymbol{T}_i,\boldsymbol{L}_c^{(i)}} \sum_i (\| \boldsymbol{T}_i - \boldsymbol{L}_c^{(i)} \boldsymbol{X}_i \|_F^2 + \| \boldsymbol{T}_i - \boldsymbol{L}_c^{(i)} (\boldsymbol{x}_i \boldsymbol{e}_k^T) \|_F^2) \tag{4}$$

As the second part of the model, to align the smoothed local structures, we have:

$$\begin{aligned} \min_{\boldsymbol{T}} \quad & \sum_i \| \boldsymbol{T}\boldsymbol{S}_i - \boldsymbol{T}_i \|_F^2 \\ \text{s. t.} \quad & \boldsymbol{T}\boldsymbol{T}^T = \boldsymbol{I} \end{aligned} \tag{5}$$

where $\boldsymbol{T}$ is the transformed coordinates of the global data and $\boldsymbol{S}_i$ is the data index matrix of the ith local structure in $\boldsymbol{T}$.

Now combining Eqs. (4) and (5), we obtain the final model of local subspace smoothness alignment (LSSA):

$$\begin{aligned} \min_{\boldsymbol{T},\boldsymbol{L}_c} \quad & \sum_i (\| \boldsymbol{T}\boldsymbol{S}_i - \boldsymbol{L}_c^{(i)} \boldsymbol{X}_i \|_F^2 + \| \boldsymbol{T}\boldsymbol{S}_i - \boldsymbol{L}_c^{(i)} (\boldsymbol{x}_i \boldsymbol{e}_k^T) \|_F^2) \\ \text{s. t.} \quad & \boldsymbol{T}\boldsymbol{T}^T = \boldsymbol{I} \end{aligned} \tag{6}$$

Fig. 3 gives an intuitive illustration of the local subspace smoothness alignment on a toyed dataset constructed by sine function.

Before ending this section, we briefly note the difference between LTSA [34] and our LSSA method. Particularly, the LTSA method derives the local coordinates of data transformation based on the assumption of local linearity, i.e., the linear correlation between data in the local structures:

$$\min_{\boldsymbol{\tau}_i,\boldsymbol{\tau}_{ij},\boldsymbol{L}_c^{(i)}} \sum_i \sum_j \| \boldsymbol{\tau}_{ij} - \boldsymbol{\tau}_i - \boldsymbol{L}_c^{(i)} (\boldsymbol{x}_{ij} - \boldsymbol{x}_i) \|^2 \tag{7}$$

where the local coordinates (i.e., $\boldsymbol{x}_{ij} - \boldsymbol{x}_i$) are learnt by PCA on the local structures. By comparing this with Eq. (4), we see that in our LSSA method, no assumption on the local linearity is made and a new geometric structure constraint is imposed on the model.

3.3. An efficient solution

To learn the model, we adopt an alternative optimization method, which is very efficient with closed-form solution at each step. Firstly, by fixing $\boldsymbol{T}$, [1] the objective can be decoupled, and

[1] Due to the correlation between $\boldsymbol{T}_i$ and $\boldsymbol{T}$, this is equal to fix $\boldsymbol{T}_i$.

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007

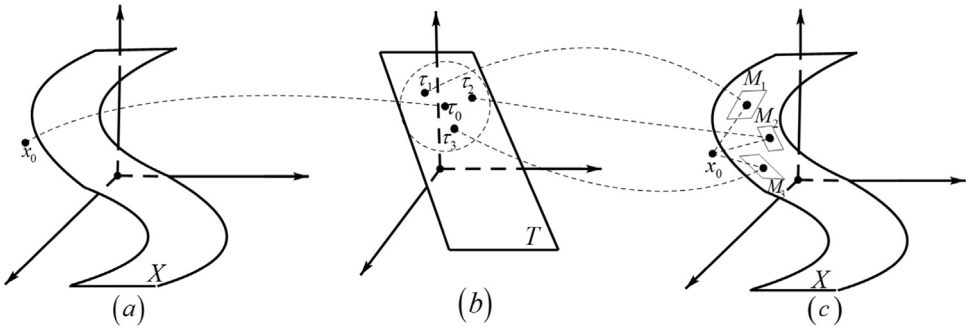


Fig. 4. Flowchart of the proposed method. See text for details.

according to Eq. (4), we have:

$$\min_{\boldsymbol{L}_c^{(i)}} \| \boldsymbol{T}_i - \boldsymbol{L}_c^{(i)}\boldsymbol{X}_i \|_F^2 + \| \boldsymbol{T}_i - \boldsymbol{L}_c^{(i)}(\boldsymbol{x}_i\boldsymbol{e}_k^T) \|_F^2 \tag{8}$$

Setting the differential of the above objective function on $\boldsymbol{L}_c^{(i)}$ to be 0 gives us the solution:

$$\boldsymbol{L}_c^{(i)} = \boldsymbol{T}_i(\boldsymbol{X}_i + \bar{\boldsymbol{X}}_i)^T(\boldsymbol{X}_i\boldsymbol{X}_i^T + \bar{\boldsymbol{X}}_i\bar{\boldsymbol{X}}_i^T)^+ \tag{9}$$

where $\bar{\boldsymbol{X}}_i$ is the matrix of the midpoints, i.e., $\boldsymbol{x}_i \times \boldsymbol{e}_k^T$.

Next, plugging the solution of $\boldsymbol{L}_c^{(i)}$ into the objective function of Eq. (6), we have

$$\sum_i (\| \boldsymbol{T}\boldsymbol{S}_i - \boldsymbol{T}_i\boldsymbol{A}_i \|_F^2 + \| \boldsymbol{T}\boldsymbol{S}_i - \boldsymbol{T}_i\boldsymbol{B}_i \|_F^2) \tag{10}$$

where $\boldsymbol{A}_i = (\boldsymbol{X}_i + \bar{\boldsymbol{X}}_i)^T(\boldsymbol{X}_i\boldsymbol{X}_i^T + \bar{\boldsymbol{X}}_i\bar{\boldsymbol{X}}_i^T)^+ \cdot \boldsymbol{X}_i$, $\boldsymbol{B}_i = (\boldsymbol{X}_i + \bar{\boldsymbol{X}}_i)^T(\boldsymbol{X}_i\boldsymbol{X}_i^T + \bar{\boldsymbol{X}}_i\bar{\boldsymbol{X}}_i^T)^+ \cdot \bar{\boldsymbol{X}}_i$. Since $\boldsymbol{T} = \boldsymbol{T}_i\boldsymbol{S}_i$, Eq. (10) can be rewritten as

$$\boldsymbol{T} \cdot (\text{trace} \sum_i \boldsymbol{S}_i((\boldsymbol{I} - \boldsymbol{A}_i)(\boldsymbol{I} - \boldsymbol{A}_i)^T + (\boldsymbol{I} - \boldsymbol{B}_i)(\boldsymbol{I} - \boldsymbol{B}_i)^T)\boldsymbol{S}_i^T) \cdot \boldsymbol{T}^T \tag{11}$$

Now, denoting $\boldsymbol{C} = \sum_i (\boldsymbol{S}_i((\boldsymbol{I} - \boldsymbol{A}_i)(\boldsymbol{I} - \boldsymbol{A}_i)^T + (\boldsymbol{I} - \boldsymbol{B}_i)(\boldsymbol{I} - \boldsymbol{B}_i)^T)\boldsymbol{S}_i^T)$, we finally reach the following objective for $\boldsymbol{T}$ by reformulating Eq. (6) as

$$\min_{\boldsymbol{T}} \quad \text{trace}(\boldsymbol{T}\boldsymbol{C}\boldsymbol{T}^T)$$
$$\text{s. t.} \quad \boldsymbol{T}\boldsymbol{T}^T = \boldsymbol{I} \tag{12}$$

Using the Rayleigh quotient [40], $\boldsymbol{T}$ could be easily derived by calculating the eigenvectors of $\boldsymbol{C}$.

3.4. Out-of-sample embedding

Due to the explicit projection functions of LSSA, it is relatively easy to embed a new sample into the manifold space. Specifically, since each point is the midpoint of a local subspace in the original feature space and each local subspace is corresponding to a local affine transformation L_c, we just need to first find the nearest point $\boldsymbol{x}_k$ for the query data $\boldsymbol{x}_0$ in the original feature space:

$$\boldsymbol{x}_k = \underset{\boldsymbol{x}_i}{\text{argmin}} \| \boldsymbol{x}_0 - \boldsymbol{x}_i \|, \quad i = 1, 2, \ldots \tag{13}$$

where $\{\boldsymbol{x}_i, i = 1, 2, \ldots\}$ are data points in the original feature space. Then we simply embed $\boldsymbol{x}_0$ using the corresponding local affine transformation $\boldsymbol{L}_c^{(k)}$ of $\boldsymbol{x}_k$:

$$\boldsymbol{\tau}_0 = \boldsymbol{L}_c^{(k)} \times \boldsymbol{x}_0 \tag{14}$$

where $\times$ denotes the projection operation. The output $\boldsymbol{\tau}_0$ is the coordinate of the query sample in manifold space.

4. Face alignment with an ensemble of local subspaces

In this section, we show how to apply the proposed method for face alignment, which effectively improves the robustness of CLM fitting compared to the traditional PDM model. Let us denote $\boldsymbol{x}_0$ the shape of points formed by concatenating the locations of facial key points estimated with discriminative detectors, then one of the most important components in a face alignment system is to verify whether this newly estimated shape of $\boldsymbol{x}_0$ is a valid "face" shape, and further to recommend a better one based on it using the face shape model learnt before.

In the traditional point distribution model (PDM), this is done with a bunch of global eigen-shapes learnt from PCA, suffering from the oversimplified hypothesis of linear shape space. Later, Yu et al. [41] proposed a new method named local coordinate coding (LCC), which reconstructs the shape of interest using samples from its neighborhood. Unfortunately, due to the noises in the initially

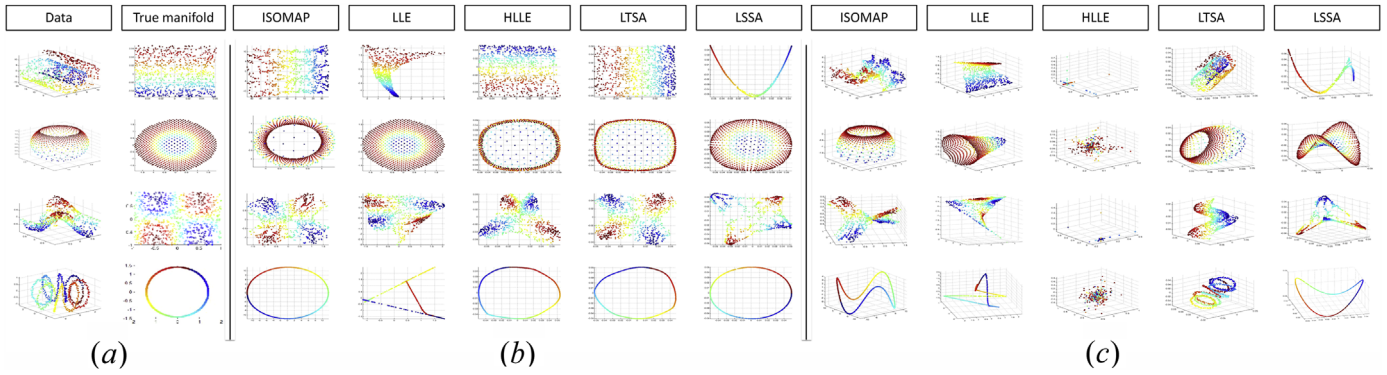


Fig. 5. Visualization of the manifolds learnt using various methods, including ISOMAP, LLE, HLLE, LTSA and ours (LSSA), one for each column, where (a) shows the original data and the corresponding ground truth; (b) shows the manifolds learnt in the inherent dimensionality; and (c) shows the manifolds learnt in the original space, while the data used are respectively Swiss roll, punctured sphere, twin peaks and toroidal helix (by row, from top to bottom).

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007

Table 1
Comparison of the running time of various manifold learning methods.

Structure	Methods			
	ISOMAP (s)	HLLE (s)	LLE (s)	LSSA (s)
Swiss roll	13.95	1.17	0.18	0.39
Punctured sphere	13.84	1.01	0.17	0.31
Twin peaks	14.58	0.98	0.20	0.26
Toroidal helix	14.70	1.03	0.15	0.31

estimated query facial shape, the single supporting local subspace used by LCC tends to be unreliable.

To address this issue, we propose to extend Yu et al.'s LCC method by simultaneously finding an ensemble of correlated local subspaces in the manifold space for the query, and then removing the redundancy of deformations using the method of sparse representation. However, directly identifying these correlated local manifolds in the original feature space suffers from the influence of noise. Instead, a three-step strategy is adopted here, as illustrated in Fig. 4 – first, we project the query shape $\boldsymbol{x}_0$ into the local manifold indexed by its nearest neighbor in the original space (Fig. 4a and b). Next, we find K nearest neighbors $\{\boldsymbol{\tau}_1, \ldots, \boldsymbol{\tau}_K\}$ in that local subspace (Fig. 4b). These neighbors serve as the robust indexes of K most correlated local subspaces $\{\boldsymbol{M}_1, \ldots, \boldsymbol{M}_K\}$ of $\boldsymbol{x}_0$ (Fig. 4b and c). Finally, we construct the deformation basis matrix $\boldsymbol{\Psi}$ using samples in these local subspaces (Fig. 4c).

With these, we can reconstruct a new face shape $\boldsymbol{x}$ for the query shape $\boldsymbol{x}_0$ using the method of sparse representation:

$$\min_{s,\boldsymbol{R},\boldsymbol{q},\boldsymbol{t}} \quad \lambda \| \boldsymbol{q} \|_1 + \| \boldsymbol{x} - \boldsymbol{x}_0 \|^2$$
$$\text{s. t.} \quad \boldsymbol{x} = s\boldsymbol{R}(\bar{\boldsymbol{x}} + \boldsymbol{\Psi}\boldsymbol{q}) + \mathbf{t} \tag{15}$$

where $\bar{\boldsymbol{x}}$ is the mean shape of training facial shapes, $\boldsymbol{\Psi}$ is the deformation matrix obtained, λ is the regularization parameter on the sparseness.

The above procedure of local subspace smoothness alignment based CLM fitting is summarized in Algorithm 1.

Algorithm 1. LSSA for CLM fitting.

Input:

The facial shapes $\boldsymbol{X}$ of training face images. The query facial shape $\boldsymbol{x}_0$ of the unseen face image.

Steps:

1. Learning the manifold space of shape vectors and projection functions $\boldsymbol{L}_c$ via Eq. (6), and embedding training shapes $\boldsymbol{X}$ into the manifold, denoted as $\boldsymbol{T}$.

2. Projecting the query shape $\boldsymbol{x}_0$ into the manifold space (cf. Eqs. (14) and (13)). Denote the resulting image as $\boldsymbol{\tau}_0$ and find its K adjacent points $\{\boldsymbol{\tau}_1, \ldots, \boldsymbol{\tau}_K\}$ among $\boldsymbol{T}$ on that manifold.

3. Find the pre-images of $\{\boldsymbol{\tau}_1, \ldots, \boldsymbol{\tau}_K\}$ in original feature space. Denote them as $\{\boldsymbol{x}_1^0, \ldots, \boldsymbol{x}_K^0\}$, each of which indexes a local subspace M_k in original feature space. Construct the deformation basis matrix $\boldsymbol{\Psi}$ using samples in these local subspaces.

4. Solving Eq. (15) to get the optimal values of parameters and using these to estimate a new facial shape $\boldsymbol{x}$ for $\boldsymbol{x}_0$. Output $\boldsymbol{x}$ as the recommended face shape.

5. Experiments

In this section, we present our experiments on two tasks. The first one is on manifold learning, in which we compare the proposed method with several classic manifold learning methods. Then we applied our method on the task of face alignment, and verified its performance on two challenging face databases, i.e., LFPW database [21] and LFW database [42].

5.1. Experiments on manifold learning

In order to verify the performance of the proposed LSSA method, we first compare our method with several representative manifold learning methods using simulated data. Specifically, three kinds of simulated data are used here: (1) the geometry structure data—Swiss roll [43] and twin peaks, (2) the sparse and

(a) LFPW (b) LFW

Fig. 6. Example image of the LFPW database (a) and the LFW database (b). The original images are in color but are shown in gray here for better visualization. Note that images in the LFPW database have higher quality than those in the LFW database.

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007

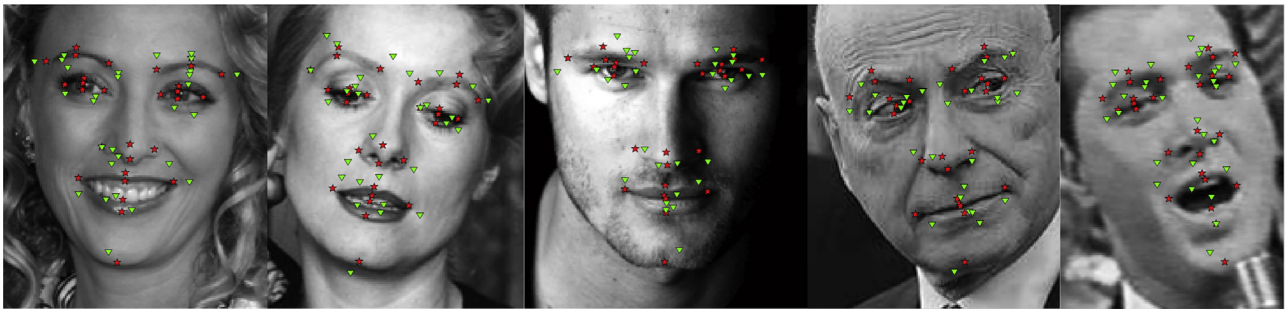


Fig. 7. Examples of perturbed ground truth of test images of the LFPW database. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

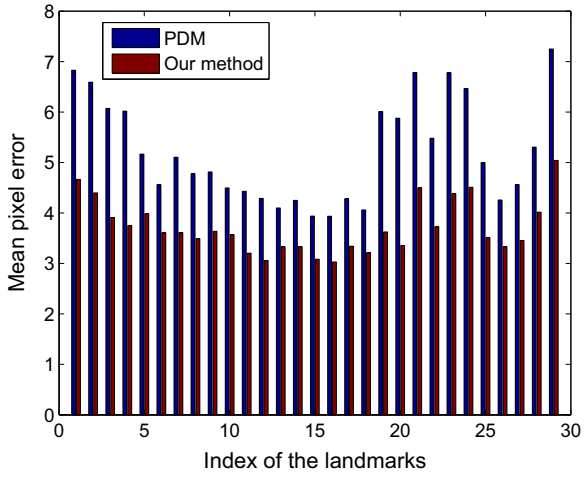


Fig. 8. Comparison of the fitting performance of two methods on the simulated data.

non-uniform sampled data—punctured sphere, (3) the noised and non-uniform sampled data—toroidal helix. The methods for comparison include: (1) the distance-preserved methods– ISOMAP [28], (2) smoothness on the second-order difference of projection–HLLE [44], (3) the local-linearity-preserved methods—LLE [32] and LTSA [34]. For experiments, we randomly generate 800 points from each type of data and the number of nearest neighbors in each local structure is set to be 8.

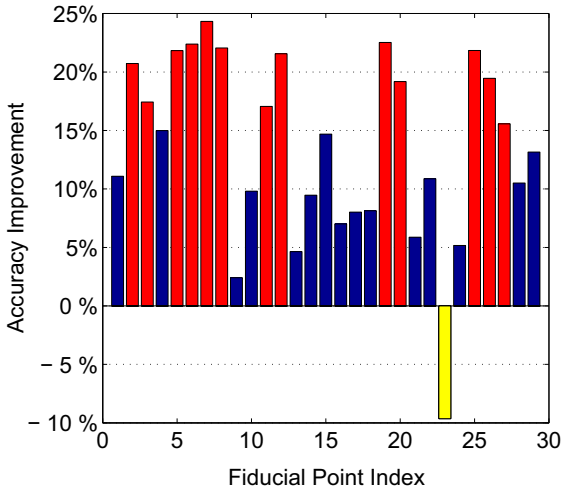
Fig. 5b gives the results. It shows that most of the compared methods manage to reveal reasonable geometric characteristics of the original data, provided that the inherent dimensionality of the data is given. However, for some data (e.g., the twin peaks, cf. the third row of Fig. 5) it is still difficult to embed them without local/global distortions. In addition, the information about the inherent dimensionality is seldom known to us in the real world, and few research investigates this problem.

Hence, in the second series of experiments, we repeated the above experiments but doing these in the original feature space, to see how these algorithms behave when the information about the inherent dimensionality is blind to us. One of the advantage of doing manifold embedding directly in the original feature space is that it effectively bypasses the difficulty of tuning the hyper-parameters and the inevitable risks of information loss during dimension reduction.

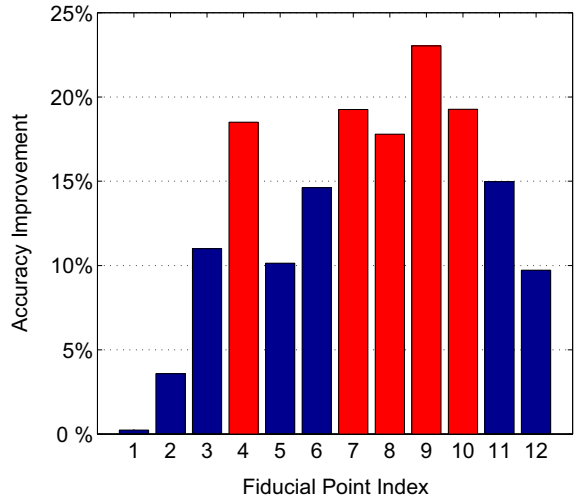
Fig. 5c gives the results. One can see that the performance of these methods drops to various extends. Particularly, the embedding data of the HLLE method is completely scattered, indicating that this method actually fails in finding any interesting structures from the data. The LLE method is better than HLLE, but it seems to shear the data too much. The ISOMAP method works well on the toroidal helix data, but not so good on the Swiss roll and punctured sphere. For the LTSA method, the structures of all the data are almost the same as those in the original data. This could partly

Fig. 9. Illustration of the located fiducial points using different method, where the first row is from the PDM method and the second row is from proposed method.

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007



(a) Performance improvement on LFPW database



(b) Performance improvement on LFW database

Fig. 10. Performance improvement over the standard constrained local model (CLM) [14] on the LFPW dataset (a) and the LFW dataset (b) by our method, where red means improvement by more than 15.0%, blue between 0.0% and 15.0%, and yellow means lower accuracy. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

be explained by the fact that it mainly learns rotation as its major transformation when working on the original feature space.

The figure reveals that overall our method works the best among the compared ones, especially on the data of punctured sphere and toroidal helix. Although the manifolds of the Swiss roll data are compressed by our method (since it is not designed to be distance-preserved), the contour of the manifold is successfully found out. But for the data with the structure of two peaks, the manifold cannot be derived well by our method and other compared methods as well.

Another attractive aspect of the proposed LSSA method lies in its efficiency, in the sense that we do not need to calculate the geodesic distance or Hessian matrix during learning. Table 1 lists the running time of different methods on a laptop computer with a 2.83 GHz GPU and 8.0 GB RAM. One can see that local smoothing methods like LLE and ours are much more efficient than the global methods such as ISOMAP and HLLE. Although our method is not as fast as the LLE method, our method yields better manifold than the LLE method, as shown in Fig. 5.

5.2. Experiments on facial landmarks localization

In this section, we present our experimental results on the task of face alignment (or facial landmarks localization). We first present our results on two popular datasets for this task and compare them with those of the state of the art methods under the CLM framework. Next we focus on investigating the usefulness of the LSSA-based CLM fitting by separating it from the effects of components for facial feature detection. Finally, we give some discussions about our method for constructing an ensemble of correlated local subspaces described in Section 4.

5.2.1. Data and settings

Two popular databases for face alignment, i.e., LFPW database (labeled face parts in the wild [21]) and LFW database (labeled face in the wild [42]) are adopted in our experiments. The images of the LFPW database [21] are collected from internet and contain large variations in pose, illumination, expression, occlusion and so on. Each image contains 29 fiducial points. Due to the null URLs of some images, 833 of the 1100 training images and 232 of the 300 test images are used in our experiments. The original LFW database [45] contains 13 233 low-resolution face images of 5749 subjects collected from web. It is originally used for face recognition and verification. In order to make it is available for face alignment, 10 fiducial points are labeled for each face images by Dantone et al. [42]. We add two more landmarks on the centers of eyes for each images in our experiments to calculate the inter-ocular distance. Example images with fiducial points of the above two datasets are respectively demonstrated in Fig. 6.

Local detector is an important component of CLMs. In our experiments, we use the popular strategy of "SIFT + SVM" to train local detectors. Additionally, we augmented the training images by left–right flip and random rotations so that we have about 6000 training images for each database. Unless otherwise noted, we use the RANSAC method [21] for initialization. The spareness parameters λ (cf. Eq. (15)) are set to be 0.01 and 0.005 respectively on the two datasets. The neighbor number in each local structure is 8, and the number of adjacent local structures are set to be 50 for the LFPW dataset and 100 for the LFW dataset respectively.

In most of the following experiments, we use the normalized root-mean-squared error (NRMSE) relative to the ground truth as the error measurement. The NRMSE is computed by dividing the root mean squared error by the distance between the two eye centers.

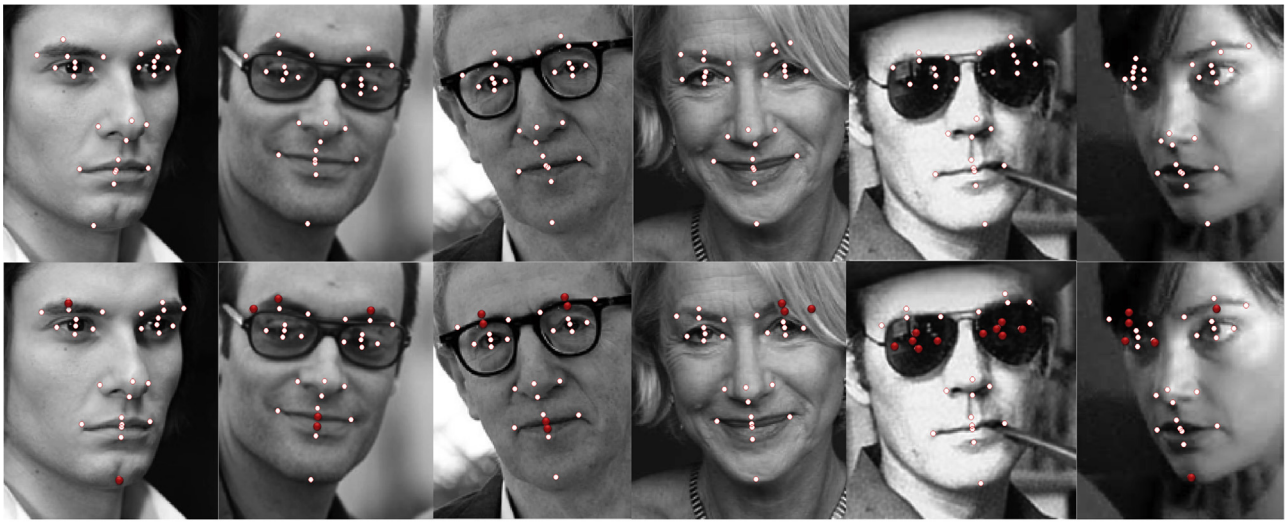
5.2.2. Contribution of the LSSA manifold learning

In this first series of experiments, we focus on verifying the effectiveness of the proposed LSSA manifold learning. Since the task of face alignment is complex and consists of several steps, we separate the component of CLM fitting from other components (e.g., facial feature detection) of the whole system by simulating the response maps and using them for LSSA learning. Specifically, we perturb the ground truth facial shapes of test images by adding Gaussian noise with mean 0 and standard deviation 10, and use these to simulate the response of the local facial feature detection. Finally, we respectively apply two fitting methods, i.e., the traditional PCA-based PDM method and the proposed LSSA method, over these simulated responses to reconstruct the corresponding improved facial shapes.

Fig. 7 demonstrates some perturbed facial shapes, where the red stars are the ground truth and the green triangles are the perturbed facial landmarks. It can be observed that some of the perturbed locations of facial points are quite far from the ground truth. This significantly increases the difficulty of CLM fitting.

To measure the performance intuitively, instead of normalized root-mean-squared error (NRMSE), in this series of experiments

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007



(a) the LFPW database

(b) the LFW database

Fig. 11. Illustration of the aligned face images from the LFPW database (a) and the LFW database (b) respectively. In each figure images in the first row are from the PDM method while those for the second row from the proposed method. Some of the most improved landmarks using our method compared to the PDM are marked with red dots. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

Table 2
Comparative performances measured by normalized root mean square error (NRMSE) (%) on LFPW.

Algorithm	NRMSE
Consensus of Examplars (CoE) [21] (our implementation)	4.30
Discriminative response map fitting (DRMF) [20]	4.68
Optimized part mixtures (OPM) [46]	6.65
Exemplar-based Graph Matching (EGM) [47]	3.98
CLM baseline with RANSAC initialization	4.96
LTSA based CLM fitting	4.83
LSSA based CLM fitting (proposed)	4.14

Table 3
Comparative performance measured by normalized root mean square error (NRMSE) (%) on LFW.

Algorithm	NRMSE
Consensus of Examplars (CoE) [21] (our implementation)	5.95
Discriminative response map fitting (DRMF) [20]	6.23
Optimized part mixtures (OPM) [46]	7.27
CLM baseline with RANSAC initialization	6.1
LTSA based CLM fitting	5.57
LSSA based CLM fitting (proposed)	5.31

we use the mean pixel error of each landmark between the estimated shapes and ground truth, defined as $err(i) = \frac{1}{N}\sum_j \| \boldsymbol{x}_j^i - \boldsymbol{o}_j^i \|_2$, where $\boldsymbol{x}_j^i$ is the estimated 2D coordinate of j-th landmark in the i-th face images, $\boldsymbol{o}_j^i$ is the corresponding coordinate of ground truth, and N is the number of test images. The sparseness parameter λ in Eq. (15) is set to be 0.01, and the number of neighbors in local structures is 8, the number of adjacent local structures is 20.

Fig. 8 gives the results. It shows that compared with the

Table 4
Comparison of different neighborhood finding strategies for CLM fitting. Performance measured with NRMSE.

Strategy	Dataset	
	LFPW	LFW
Estimation via SR in manifold space	5.21	6.58
Estimation via SR in original space	4.53	5.47
The proposed method	4.14	5.31

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007

original PDM model, our method yields better fitting accuracy consistently over all of the 29 fiducial points defined in the LFPW dataset. Particularly, the mean error of ours is 3.7 pixels, compared to 5.2 pixels of the PDM model. This clearly demonstrates that our method behaves more robustly against large feature detection errors. Fig. 9 illustrates 29 located fiducial points with white dots by the two methods in some face images. It shows that although not all fiducial points are aligned well using our method due to the large perturbations, it works much better than the PDM method.

5.2.3. Comparison with the baseline algorithm

First we compare our algorithm with the baseline algorithm, i.e., the standard constrained local model (CLM) [14]. Both algorithms share the same face alignment pipeline and the same sets of local feature detectors, and the difference is that the CLM's face shape vector updating component is based on PCA while ours is based on LSSA as described in Section 4.

Fig. 10 gives the results. The figure shows that our method significantly outperforms the baseline algorithm. Particularly, by replacing the linear subspace of CLM with the local manifolds learnt with LSSA, our algorithm improves the CLM by 15.0% over 44.8% of the 29 fiducial points on the LFPW dataset. By checking the distribution of the locations of these mostly improved fiducial points, we find that among others eyebrows (index 2,3,5–8), medial angle of eyelid (11,12), wing of nose (19,20) and lips (25–27) benefit most. Since the appearance of these facial points is easy to be varying, the geometrical support from nonlocal parts become more crucial, which partly explains why our algorithm is effective in dealing with these facial regions. While for the remaining fiducial points, it seems that the performance of local facial detectors plays more important role than the shape constraints during face alignment, and our method improves by a small margin on them.

Fig. 11 gives some illustration of the aligned face images from both datasets, where the landmarks of the images in the first row are predicted by the PCA-based traditional CLM method and the ones in the second row are from our algorithms. It shows that our method performs more reliable than the CLM method especially on those facial points with large appearance or geometric variations, such as middle of lips, canthus, chin and eyebrows for LFPW additionally.

5.2.4. Comparison with the state of the art CLM variants

Next we compare our method with several state of the art algorithms. Most of these algorithms can be understood as variants of the traditional CLM methods with different enhancement. For example, the optimized part mixtures (OPM) [46] focuses on modeling the distribution of facial parts instead of that of the global shape vectors, discriminative response map fitting (DRMF) [20] improves the CLM by discriminatively learning from response maps, and both Consensus of Examplars (CoE) [21] and its variant —Exemplar-based Graph Matching (EGM) [47] can be thought of as nonparametric CLM models. In addition, we also compare our method with another manifold-based CLM fitting method by replacing the PCA component with the LTSA. The dimensionality of LTSA is tuned to be optimal over the validation set and the whole training procedure and other experimental settings are kept the same as those in our method.

Tables 2 and 3 respectively give the comparative results on the LFPW dataset and the LFW dataset. These tables reveal that, although among these CLM variants our method is possibly the most simplest one by simply replacing the PCA with the proposed LSSA manifold learning, we achieve comparable or preferable performance with the state of the art variants of CLM methods. Particularly, on the LFPW dataset, our method performs better than most of the compared ones except the EGM [47], while on the LFW dataset, our method performs the best,[2] with improvement by 0.64% over the CoE method in terms of the NRMSE value. The tables also show that our method works consistently better than the LTSA method on both datasets.

5.3. Discussions

In Section 4, we mention that to find an ensemble of correlated local subspaces in the manifold space for a query shape, we propose a three-step strategy. Here we empirically compare this with another two optional strategies to find the neighborhood in the algorithm of CLM fitting:

- Estimate neighborhood facial shapes in the manifold for the query using sparse reconstruction (SR) and then project back the reconstructed point into original space.
- Directly find the neighborhood in the original feature space using sparse reconstruction (SR).

Table 4 gives the comparative performance using various strategies on the LFPW dataset and the LFW dataset. It can be seen that the proposed three-step strategy performs the best on both datasets. Note that theoretically the second alternative strategy is very similar to ours but the chosen data for reconstruction are not local enough, while for the first strategy, since the manifold embedding (LSSA here) is not distance-preserved, the scaling variety could damage the robustness. This partly explains why we should reconstruct shape vector in the original feature space. Actually, our method chooses the neighborhood for the query in the manifold but do the reconstruction in the original space, which effectively combines the best of both worlds.

6. Conclusion

In this paper, we present a novel manifold based constrained local model fitting named local subspace smoothness alignment (LSSA). The LSSA method learns the manifold in the original dimensionality with a new geometric measurement for the curvature of local structures. Based on the learnt manifold, we introduce an improved face alignment method under the framework of constrained local model (CLM). It performs robust CLM fitting in the original feature space but using adjacent deformations of the query found in the manifold space, hence effectively combining the best of both worlds. We demonstrate the effectiveness of the proposed method on two challenging face alignment datasets with encouraging results.

Acknowledgments

We thank the anonymous reviewers for their in-depth comments, suggestions, and corrections, which have greatly improved the paper. This work is partially supported by National Science Foundation of China (61373060) and Qing Lan Project.

References

[1] D.D. Lee, H.S. Seung, Learning the parts of objects by non-negative matrix factorization, Nature 401 (6755) (1999) 788–791.
[2] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.

[2] Unfortunately the results of the EGM [47] method on the LFW dataset are not available in their paper.

[3] S. Ren, X. Cao, Y. Wei, J. Sun, Face alignment at 3000 fps via regressing local binary features, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, 2014, pp. 1685–1692.
[4] X. Cao, Y. Wei, F. Wen, J. Sun, Face alignment by explicit shape regression, Int. J. Comput. Vis. 107 (2) (2014) 177–190.
[5] D. Lee, H. Park, C.D. Yoo, Face alignment using cascade gaussian process regression trees, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4204–4212.
[6] S. Zhu, C. Li, C.C. Loy, X. Tang, Face alignment by coarse-to-fine shape searching, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 4998–5006.
[7] Y. Sun, Q. Liu, H. Lu, et al., Low rank driven robust facial landmark regression, Neurocomputing 151 (2015) 196–206.
[8] Y. Yang, Y. Su, D. Cai, M. Xu, Nonlinear deformation learning for face alignment across expression and pose, *Neurocomputing* 195 (2016) 149–158, http://dx.doi.org/10.1016/j.neucom.2015.08.114.
[9] Z. Cui, S. Shan, H. Zhang, S. Lao, X. Chen, Image sets alignment for video-based face recognition, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Rhode island, 2012, pp. 2626–2633.
[10] V. Le, J. Brandt, Z. Lin, L. Bourdev, T.S. Huang, Interactive facial feature localization, in: Computer Vision—ECCV 2012, Springer, Firenze, 2012, pp. 679–692.
[11] G. Tzimiropoulos, Project-out cascaded regression with an application to face alignment, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3659–3667.
[12] Y. Zhang, Z. Tang, C. Zhang, J. Liu, H. Lu, Automatic face annotation in tv series by video/script alignment, Neurocomputing 152 (2015) 316–321.
[13] T.F. Cootes, C.J. Taylor, D.H. Cooper, J. Graham, Active shape models – their training and application, Comput. Vis. Image Underst. 61 (1) (1995) 38–59.
[14] D. Cristinacce, T.F. Cootes, Feature detection and tracking with constrained local models, in: BMVC, vol. 1, Citeseer, 2006, p. 3.
[15] K. Nickels, S. Hutchinson, Estimating uncertainty in ssd-based feature tracking, Image Vis. Comput. 20 (1) (2002) 47–58.
[16] X.S. Zhou, A. Gupta, D. Comaniciu, An information fusion framework for robust shape tracking, IEEE Trans. Pattern Anal. Mach. Intell. 27 (1) (2005) 115–129.
[17] Y. Wang, S. Lucey, J.F. Cohn, Enforcing convexity for improved alignment with constrained local models, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), IEEE, Dayton, 2008, pp. 1–8.
[18] L. Gu, T. Kanade, A generative shape regularization model for robust face alignment, in: Computer Vision—ECCV 2008, Springer, Marseille, 2008, pp. 413–426.
[19] J.M. Saragih, S. Lucey, J.F. Cohn, Deformable model fitting by regularized landmark mean-shift, Int. J. Comput. Vis. 91 (2) (2011) 200–215.
[20] A. Asthana, S. Zafeiriou, S. Cheng, M. Pantic, Robust discriminative response map fitting with constrained local models, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Portland, 2013, pp. 3444–3451.
[21] P.N. Belhumeur, D.W. Jacobs, D.J. Kriegman, N. Kumar, Localizing parts of faces using a consensus of exemplars, IEEE Trans. Pattern Anal. Mach. Intell. 35 (12) (2013) 2930–2940.
[22] S. Cheng, S. Zafeiriou, A. Asthana, M. Pantic, 3d facial geometric features for constrained local model, in: 2014 IEEE International Conference on Image Processing (ICIP), IEEE, Paris, 2014, pp. 1425–1429.
[23] P. Martins, R. Caseiro, J.F. Henriques, J. Batista, Likelihood-enhanced Bayesian constrained local models, in: 2014 IEEE International Conference on Image Processing (ICIP), IEEE, Paris, 2014, pp. 303–307.
[24] T.F. Cootes, G.J. Edwards, C.J. Taylor, Active appearance models, IEEE Trans. Pattern Anal. Mach. Intell. (6) (2001) 681–685.
[25] X. Cao, Y. Wei, F. Wen, J. Sun, Face alignment by explicit shape regression, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Rhode island, 2012, pp. 2887–2894.
[26] X. Xiong, F. De la Torre, Supervised descent method and its applications to face alignment, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Portland, 2013, pp. 532–539.
[27] V. Kazemi, S. Josephine, One millisecond face alignment with an ensemble of regression trees, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Columbus, 2014.
[28] J.B. Tenenbaum, V. De Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[29] S. Yan, D. Xu, B. Zhang, H.J. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Trans. Pattern Anal. Mach. Intell. 29 (1) (2007) 40–51.
[30] I. Borg, P.J. Groenen, Modern Multidimensional Scaling: Theory and Applications, Springer Science & Business Media, New York, 2005.
[31] T. Zhang, D. Tao, X. Li, J. Yang, Patch alignment for dimensionality reduction, IEEE Trans. Knowl. Data Eng. 21 (9) (2009) 1299–1313.
[32] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[33] X. Niyogi, Locality preserving projections, in: Neural Information Processing Systems, vol. 16, MIT, Calcutta, 2004, p. 153.
[34] Z.-y. Zhang, H.-y. Zha, Principal manifolds and nonlinear dimensionality reduction via tangent space alignment, J. Shanghai Univ. (Engl. Ed.) 8 (4) (2004) 406–424.
[35] P. Zhang, H. Qiao, B. Zhang, An improved local tangent space alignment method for manifold learning, Pattern Recognit. Lett. 32 (2) (2011) 181–189.
[36] R. Chatpatanasiri, B. Kijsirikul, A unified semi-supervised dimensionality reduction framework for manifold learning, Neurocomputing 73 (10) (2010) 1631–1640.
[37] H. Qiao, P. Zhang, D. Wang, B. Zhang, An explicit nonlinear mapping for manifold learning, IEEE Trans. Cybern. 43 (1) (2013) 51–63.
[38] D. Lunga, S. Prasad, M.M. Crawford, O. Ersoy, Manifold-learning-based feature extraction for classification of hyperspectral data: a review of advances in manifold learning, IEEE Signal Process. Mag. 31 (1) (2014) 55–66.
[39] Q. Wang, W. Wang, R. Nian, B. He, Y. Shen, K.-M. Björk, A. Lendasse, Manifold learning in local tangent space via extreme learning machine, Neurocomputing 174 (2016) 18–30, http://dx.doi.org/10.1016/j.neucom.2015.03.116.
[40] K.-J. Bathe, Finite Element Procedures, Klaus-Jurgen Bathe, New Jersey, 2006.
[41] K. Yu, T. Zhang, Y. Gong, Nonlinear learning using local coordinate coding, in: Advances in Neural Information Processing Systems, 2009, pp. 2223–2231.
[42] M. Dantone, J. Gall, G. Fanelli, L. Van Gool, Real-time facial feature detection using conditional regression forests, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Rhode island, 2012, pp. 2578–2585.
[43] J.B. Tenenbaum, Mapping a manifold of perceptual observations, in: Advances in Neural Information Processing Systems, 1998, pp. 682–688.
[44] D.L. Donoho, C. Grimes, Hessian eigenmaps: locally linear embedding techniques for high-dimensional data, Proc. Natl. Acad. Sci. 100 (10) (2003) 5591–5596.
[45] G.B. Huang, M. Ramesh, T. Berg, E. Learned-Miller, Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments, Technical Report 07-49, University of Massachusetts, Amherst, October 2007.
[46] X. Yu, J. Huang, S. Zhang, W. Yan, D.N. Metaxas, Pose-free facial landmark fitting via optimized part mixtures and cascaded deformable shape model, in: 2013 IEEE International Conference on Computer Vision (ICCV), IEEE, Sydney, 2013, pp. 1944–1951.
[47] F. Zhou, J. Brandt, Z. Lin, Exemplar-based graph matching for robust facial landmark localization, in: 2013 IEEE International Conference on Computer Vision (ICCV), IEEE, Sydney, 2013, pp. 1025–1032.

Dakun Liu was born in 1984. He received his BS and MS degrees in applied mathematics in 2006 and 2009 respectively. He received his Ph.D. degree in computer science and technology from Nanjing University of Aeronautics and Astronautics in 2016. Now he is a lecturer in Yancheng Institution of Technology. His research interests include computer vision, machine learning, etc.

Xiaoyang Tan was born in 1971. He received the Ph.D. degree in machine learning from Nanjing University in 2005. He is a professor and Ph.D. supervisor at Nanjing University of Aeronautics and Astronautics. His research interests include computer vision, machine learning, etc.

Please cite this article as: D. Liu, X. Tan, Local subspace smoothness alignment for constrained local model fitting, Neurocomputing (2016), http://dx.doi.org/10.1016/j.neucom.2016.07.007