Signal Processing 112 (2015) 34–42

Contents lists available at ScienceDirect

Signal Processing

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

Supervised sparse manifold regression for head pose estimation in 3D space

Qicong Wang [a], Yuxiang Wu [a], Yehu Shen [b], Yong Liu [a], Yunqi Lei [a,*]

[a] Department of Computer Science, Xiamen University, Xiamen 361005, China
[b] Department of System Integration and IC Design, Suzhou Institute of Nano-Tech and Nano-Bionics, CAS, Shuzhou 215125, China

ARTICLE INFO

Article history:
Received 15 March 2014
Received in revised form
16 July 2014
Accepted 17 July 2014
Available online 24 July 2014

Keywords:
Manifold learning
Supervised learning
Sparse regression
Head pose estimation

ABSTRACT

In estimating the head pose angles in 3D space by manifold learning, the results currently are not very satisfactory. We need to preserve the local geometry structure effectively and need a learned projective function that can reveal the dominant features better. To address these problems, we propose a Supervised Sparse Manifold Regression (SSMR) method that incorporates both the supervised graph Laplacian regularization and the sparse regression into manifold learning. In SSMR, on the one hand, a low-dimensional projection is embedded to represent intrinsic features by using supervised information while the local structure can be preserved more effectively by using the Laplacian regularization term in the objective function. On the other hand, by casting the problem of learning projective function into a regression with L_1 norm regularizer, a projection is mapped to carry out the sparse representation of high dimension features, rather than a compact linear combination, so as to describe the dominant features better. Experiments show that our proposed method SSMR is beneficial for head pose angle estimation in 3D space.

1. Introduction

Head pose estimation is an important preprocess necessary for face recognition, human object tracking, and human–machine interface [1]. In the past few years, although the face tracking or face recognition related researches have achieved great progresses, works on robust algorithms for changing head pose are demanded and there still remain difficulties to be overcome. Since its accuracy is prerequisite and crucial to improve the performance of the face related problems, head pose estimation has been paid more and more attentions [2,3]. Specifically, the task of head pose estimation is to determine the head pose angles in 3D space from the input 2D face images. The rotation of the head has three types: yaw, pitch and roll. The rotation angles are between [−90°, 90°] of each type, and the result of head pose estimation is always represented as a vector. Commonly, the yaw angle estimation should be fulfilled first [4,5], and the angle estimation for other two types could use the same method to deal with.

Besides using 2D images, using of range data is an another way to estimate the 3D pose [6], yet it needs an expensive equipment to get the 3D data and it involves much more computation. The pose estimation is also one of the key tasks for 3D object retrieval and recognition. An interactive and computationally efficient 3D object retrieval scheme with query view selection approach is given by Gao et al. [7]. In [8], an approach of hypergraph analysis is proposed for 3D object retrieval and recognition, where the multiple hypergraphs are constructed for 3D objects from their 2D views. A Hausdorff distance learning scheme for interactive 3D object retrieval is provided in [9].

* Corresponding author. Tel.: +86 592 2580057.
E-mail address: yqlei@xmu.edu.cn (Y. Lei).

http://dx.doi.org/10.1016/j.sigpro.2014.07.011
0165-1684/

In recent years, manifold learning has been widely exploited in various machine learning areas such as pattern recognition, image analysis and data mining. There already exist several different manifold learning methods, for example, Locally Linear Embedding (LLE) [10], ISOMAP [11] and Laplacian Eigenmap (LE) [12], etc. As a special class of dimension reduction techniques which are important to machine learning, the manifold learning attempts to learn a low-dimensional manifold structure from the observation of high dimension space. Surely, the technique is applicable in head pose estimation. A fundamental assumption of using this technique is that the face images with changing head pose are often highly nonlinear to the pose information, then they can be naturally considered as a kind of low-dimensional manifold structure. Thus, using manifold learning to compute the head pose angle in 3D space from single 2D face image should be beneficial. The manifold models the nonlinear and continuous variations of face appearance, and if learn properly, new face images can then be embedded in a low-dimensional space to estimate the head pose angle. In [13], Raytchev et al. have proposed a kind of nonlinear pose image expression technique based on ISOMAP. Fu & Huang use Graph Embedding (GE) technique for head pose estimation [14]. In [14], they first construct neighborhood weight graph under LLE through graph embedded linearization, then the projection direction from a high-dimensional space to a low-dimensional embedding is found. Finally, the head pose category of test sample is obtained by using nearest neighbor classifier.

In [15,16], Balasubramanian et al. propose a framework of Biased Manifold Embedding (BME), where they use the head pose label information before obtaining low-dimensional embedding, then they use a Generalized Regression Neural Network (GRNN) to learn the nonlinear mapping to deal with the out-of-sample data points. Yan et al. in [17], by considering several kinds of manifold learning methods, propose an unified graph embedding and extensions framework of dimensionality reduction algorithm based on the spectral graph theory. In [18], by incorporating head pose angle information as the supervision into the objective function, BenAbdelkader et al. propose a framework of supervised manifold learning for head pose estimation, where the cubic splines smoothing and support vector regression are used to get nonlinear mapping to estimate the head pose angles in the low-dimensional embedded space. Some linearized version of Locally Linear Embedding (LLE) and Laplacian Eigenmap (LE), Neighborhood Preserving Embedding (NPE) [19] and Locality Preserving Projections (LPP) [20] are proposed by He et al. These methods are efficient, can directly handle the out-of-sample extension and can fully show the characteristics and advantages of linear manifold learning.

However, these current methods may have a drawback that the linear projection is just a compact linear combination of all the high-dimensional original features. There is an important problem that cannot be neglected, that is, from the viewpoint of feature selection and extraction, the dominant features should be maintained while others should be depressed. To handle this problem, in this paper we add a local graph Laplacian term with L_1 norm in the optimization objective of the manifold learning to get the low-dimensional embedding. Through this sparse regression minimization we can obtain the projection mapping matrix, then the sparse representation of high-dimensional features can be conducted. The supervised Laplacian regularization into objective function helps us to achieve the preservation of both the local geometry and the dominant features.

In the following sections, this paper is organized as the following: Section 2 presents the supervised manifold learning method based on the sparse regression. Section 3 provides experimental results of head pose estimation on the FacePix dataset. Section 4 is the conclusion, and some future works are also suggested.

2. Supervised manifold learning and sparse regression

2.1. Graph construction

The manifold learning can be regarded as a procedure of two stages of graph construction and graph embedding [17,21]. The graph construction should be conducted first. Specifically, for a given sample set $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,\ldots\mathbf{x}_N]$, where $\mathbf{x}_i \in \mathbb{R}^D$, $i=1,\ldots N$, the corresponding low-dimensional embedding of each sample is $\mathbf{Y}=[\mathbf{y}_1,\mathbf{y}_2,\ldots,\mathbf{y}_n]$, where $\mathbf{y}_i \in \mathbb{R}^d$, $i=1,\ldots N$, $d \ll D$. Their corresponding patterns, the head pose angles in this paper, are denoted as β_i, $i=1,\ldots N$. Quite often, the k nearest neighbor criterion is used to construct weight graph. But if just simply use Euclidean distance of the sample $\mathbf{x}_i$ and $\mathbf{x}_j$ to determine whether they are k nearest neighbors, many information may be lost. This is because in original high-dimensional feature space, the sample similarity is much more affected by image details, such as expressions and illuminations, while less by head pose information. BME [15,16] framework introduces biased distance into neighbor closure measurement to reduce deviation brought from image's L_2 norm. The biased distance is described in the form of

$$\tilde{\mathbf{D}}(i,j)=\mathbf{D}(i,j)\frac{dist(i,j)+\varepsilon}{MaxDist-dist(i,j)+\varepsilon}; \qquad i,j=1,\ldots N. \tag{1}$$

where $\mathbf{D}(i,j)=\|\mathbf{x}_i-\mathbf{x}_j\|^2$ stands for the Euclidean distance in the original feature space, and ε is a label adjusting factor; $dist(i,j)=\gamma\cdot|\beta_i-\beta_j|$, where γ is a control parameter, β_i,β_j are the head pose angles of sample $\mathbf{x}_i$, $\mathbf{x}_j$, respectively. Meanwhile, $MaxDist=\max\{dist(i,j)|i,j=1,\ldots,N\}$, it denotes the maximum distance of $dist(i,j)$ between all pairs. By Eq. (1), $\tilde{\mathbf{D}}(i,j)$ becomes smaller when the head poses of sample $\mathbf{x}_i$ and $\mathbf{x}_j$ are more similar and closer, and becomes bigger when the poses are more separated. Through such biased distance measure $\tilde{\mathbf{D}}(i,j)$, the supervised k nearest neighbor set $N_k(\mathbf{x}_i)$ of each sample $\mathbf{x}_i$ could be obtained, which can represent better the head pose distance.

Now a Gaussian kernel weight matrix $\mathbf{S}$ that is related with $\tilde{\mathbf{D}}(i,j)$ is necessary in our manifold learning.

(a) *The Gaussian kernel* [12,20]:

$$\mathbf{W}_{i,j}=\begin{cases}\exp\left(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{2\sigma^2}\right), & \mathbf{x}_i\in N_k(\mathbf{x}_j)\vee\mathbf{x}_j\in N_k(\mathbf{x}_i)\\ 0 & \text{otherwise}\end{cases} \tag{2}$$