AN EXTENSION OF LOCALLY LINEAR EMBEDDING FOR POSE ESTIMATION OF 3D OBJECT

XU ZHANG[1], HUI-MIN MA[2], YU-SHU LIU[1], CHUN-XIAO GAO[1]

[1]School of Computer Science and Technology, Beijing Institute of Technology, Beijing 100081,China
[2]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
E-MAIL: zhang_moi@163.com, mhmpub@tsinghua.edu.cn

Abstract:

Diverse pose estimation of 3D object in the whole view-space is a problem perplexed many researchers. In this paper we propose an algorithm extended from LLE which can estimate the arbitrary pose of 3D object in the whole view space. First, we compute the eigen-images of training set by introducing the idea of PCA using the low-dimensional embedding coordinate deduced from LLE. For a new sample we can compute its projection to the eigen-images, and the nearest training images from the new sample are the estimation poses. Next, we set different weight for different projection direction depends on its eigen-value when computing the distance between the new sample and the training images. Experimental results obtained demonstrated that the performance of the proposed method could estimate the diverse pose of 3D object efficiently and precisely, also our algorithm can be extended to real-time pose estimate, is of a potential future.

Keywords:

Locally linear embedding; Dimensionality reduction; Pose estimation of 3D object; Eigen-image

1. Introduction

Pose estimation has been studied in pattern recognition and computer vision fields since its beginning. Many researches have been done in the field. The field is of great importance because it is crucial for many computer and robot vision tasks, such as object grasping, manipulation and recognition or self-localizaion of mobile robots, auto-navigation of flying vehicle, etc.

Pioneering work on pose estimation was done can be broadly categorized into several below groups: shape-based geometric analysis [1] where head pose is deduced from geometric information like configurations of object landmarks, template matching [2] based on nearest neighbor classification with texture templates, graph-match method [3], gradient flow approach [4], appearance-based methods [5], and so on. Recently, a lot of new methods have been proposed, such as the probabilistic methods [6], color cooccurrence histograms [7], some variations methods based the former [8], etc.

At present, diverse pose estimation of 3D object in the whole viewpoint space is an issue perplexed many researchers and little work was done on the field as far as I know. Most applications of pose estimation concentrate on the head pose estimation [9], face pose estimation [10], human body pose estimation [11], etc. Also pose estimation of 3D object is still a new research field. Especially, one of the main difficulties in pose estimation of 3D object is how to cope with the seemingly high-dimensional space of stimuli. For example, when we sampling every 2 degree both in longitude and latitude (see figure 1), within the range from 0^{th} and 14^{th} latitude in the first octant on the first eighth sphere of the whole view space, the poses of an object will be 360, even the space of a small 350×300 gray pose-image of an object is already of dimension 37800000.

Due to this, series of dimension reduction methods are proposed, such as Principal Component Analysis (PCA) [12], Independent Component Analysis [13], Linear Discriminate Analysis [12], nonlinear analysis [14]. Recent years, manifold learning as the nonlinear analysis method has attracted much attention, as a basic manifold learning method, LLE algorithm finds a single global coordinate system of lower dimensionality and preserves neighborhood relationships, thereby reveals the underlying structure of the data.

In the paper, we propose an algorithm extended from LLE which can estimate the diverse poses of 3D object in the whole view space. In our algorithm, we present a method to compute the eigen-images of training set by introducing the idea of PCA using low-dimensional embedding coordinate deduced from LLE. For a new sample we can compute its projection to the eigen-images, and the nearest training images from the new sample are the estimation poses.

The point should be emphasized is that we set different

1-4244-0973-X/07/$25.00 ©2007 IEEE

weight for different projection direction depends on its eigen-value when computing the distance between the sample and the training images.

Experimental results obtained demonstrate that the performance of the proposed method could estimate the diverse pose of 3D object efficiently and precisely.

2. Locally Linear Embedding (LLE)

Roweis and Saul presented LLE algorithm [15] as an unsupervised learning algorithm that computes low-dimensional, neighborhood-preserving embeddings of high-dimensional inputs data. They presented LLE as a method that maps its input into a single global coordinate system of lower dimensionality. By exploiting the local symmetries of linear reconstructions, LLE is able to learn the global structure of non-linear manifolds.

In our experiments, LLE is tested on projection image of four kind of object (see figure 3), which are embedded in a low-dimensional space and demonstrate the neighborhood properties.

We restrict the description of LLE to the following three steps, which facilitate the understanding of the further analysis. More details of the implementation can be found in [15].

Let $X = \{x_1, x_2, \ldots, x_N\}$ be a training set of N points in a high-dimensional data space R^D. The data points are assumed to lie on or near a nonlinear manifold of intrinsic dimensionality $d < D$ (typically $d \ll D$). Provided that enough data could be available so that the underlying manifold can be considered to be "well-sampled", then each individual data point of this training set and its corresponding neighbors would be sufficiently close to lie within a locally linear patch on the manifold. The goal of LLE is to find a low-dimensional embedding of X by mapping the D-dimensional data into a single global coordinate system in R^d. Let us denote the corresponding set of the N points in the embedding space R^d by $Y = \{y_1, y_2, \ldots y_N\}$.

Step1 Find the set N_i of k nearest neighbors of each point X_i.

Step2 Compute the reconstruction weights of the neighbors that could best reconstruct each data point from its neighbors, minimizing the cost function in equation below

$$\varepsilon(W) = \sum_i \left| X_i - \sum_j W_{ij} Xj \right|^2 \qquad (1)$$

which subject to constraints:

(1)for each $X_j \notin N_i$, $W_{ij} = 0$

(2) the rows of the weight matrix sum to one: $\sum_j W_{ij} = 1$

step3 compute the low-dimensional embedding Y which best reconstructed by the weight matrix W, minimizing the following eq.2

$$\Phi(Y) = \sum_i \left| Y_i - \sum_j W_{ij} Y_j \right|^2 \qquad (2)$$

which subject to the two constraints: $\sum_{i=1}^{N} Y_i = 0$, and $\frac{1}{N}\sum_{i=1}^{N} Y_i Y_i^T = I$,where 0 is a column vector of zeros and I is an identity matrix.

3. Algorithm on pose estimation of 3D object

Our algorithm on diverse pose estimation of 3D object in whole view-space consists of the following steps,

Step 1, The whole date set X is divided into two subsets X_{train}, X_{test}, so that $X_{train} \cap X_{test} = \Phi$, $X_{train} \cup X_{test} = X$. For each data point $\in X_{train}$, we computing its low-dimensional embedding coordinate Y using LLE algorithm.

Step 2, Computing the eigen-image I of the low dimensional embedding based on idea of PCA, and we assume that there exist series of eigen-images in the low-dimensional embedding coordinate system as PCA method, the eigen-image is computed as

$$I_k = X^{-1} Y_k, \quad Y_k \text{ is the k}^{th} \text{ coordinate.} \qquad (3)$$

I is considered as the principal components of the training set of the model images. In next step pose estimation is performed by projecting a test image into the subspace spanned by the eigen-image and estimating the pose by comparing its position in eigen-image with the positions of known individuals.

Step 3, For a test image x_i, we computing its projection coordinate onto the eigen-image as below:

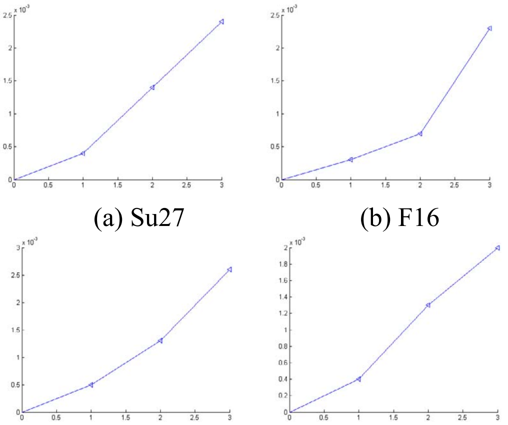
distance= $x_i \cdot I$, "·" denotes the inner-production of vector.

distance is a d-dimensional vector, *distance*(i) denote the distance between the test image and the i^{th} coordinate.

Step 4, Find the embedding coordinate holds the nearest distance to the projection coordinate from embedding coordinates as below:

$$\arg\min_j \left(\sum_{i=1}^{d} W_i \left(Y_{ij} - disance(i) \right) \right) \qquad (4)$$

subject to the constraint : $W_i = \lambda_{d-i+1} / \left(\sum_{k=1}^{d} \lambda_k \right)$, λ_i is the eigen-value corresponding to the i^{th} coordinate, and j is the index of a training image, Y_{ij} is the i^{th} projection coordinate of the j^{th} training image.



(a) Su27 (b) F16

(c) tank (d) car

Figure 1. The first three eigen-value corresponding to the low-dimensional embedding coordinate

From the computing process of LLE algorithm, we know that $\lambda_1 < \lambda_2 < \lambda_3 < \ldots$, and the smaller the eigen-value is, the more representative the corresponding eigen-vector is.

Figure 1 shows the first three eigen-value corresponding to the low-dimensional embedding coordinate system. In order to emphasize distinct eigen-image according to their importance, we apply different weight to different projection direction in a reverse order of eigen-value, i.e. $W_i = \lambda_{d-i+1} / \left(\sum_{k=1}^{d} \lambda_k \right)$.

Finally, the pose of the corresponding image of index j is considered as the pose of the test image.

4. Experiments

In order to investigate the performance of the proposed method, the Princeton Shape Benchmark (PSB) [16] is used in our experiment. PSB is a publicly available database of polygonal models collected from the World Wide Web and a suite of tools for comparing shape matching and classification algorithms. We select four class objects from PSB and produce a series of diverse pose (projection image) of the model to represent the 3D object efficiently. Here, in order to illuminate our algorithm and verify its universal, we sampling every 2 degree both in longitude and latitude (see figure 2), within the range from 0^{th} and 14^{th} latitude in the first octant on the first eighth sphere of the whole view space, for the pose of the area is various. The number of samples is set as 360 manually, and every sample represents a certain kind of pose of an object from PSB.

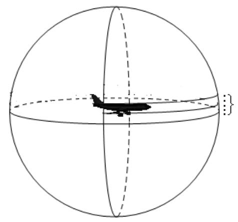
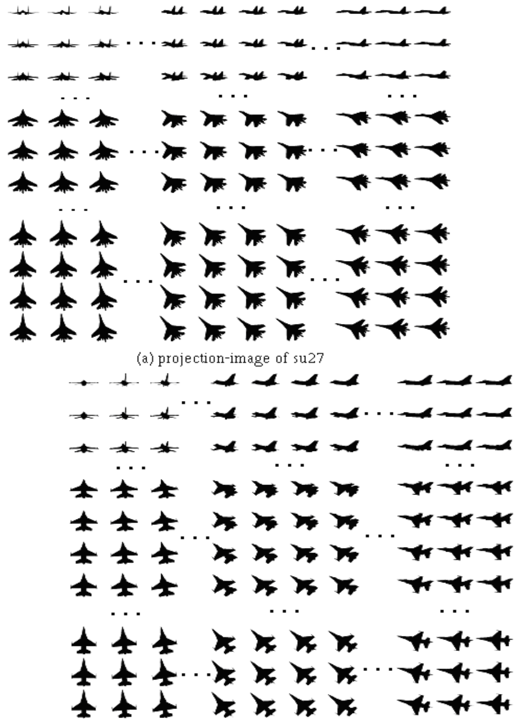


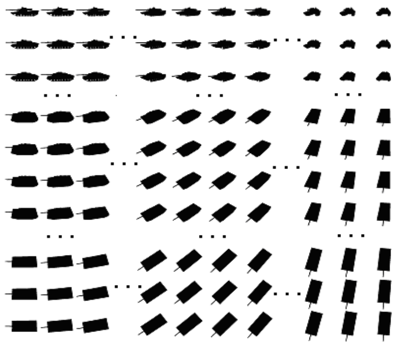
Figure 2. Sampling range of a model

Figure 3 shows part of the sequential poses of su27,f16, tank and car in the first octant on the first eighth sphere of the whole view space:

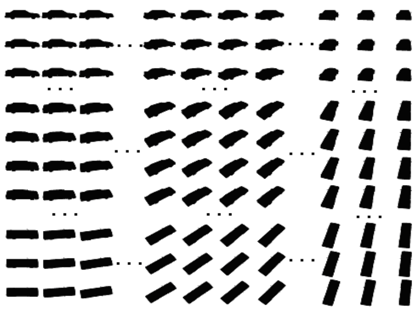


(a) projection-image of su27

(b) projection-image of F16



(c)projection-image of m1a2-tank



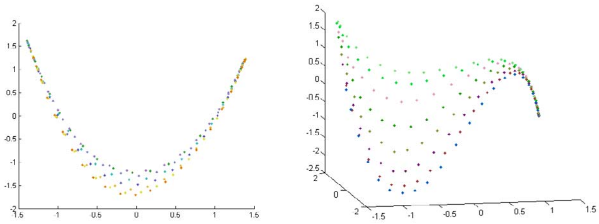
(d)projection-image of car

Figure 3. The diverse poses of su27, f16, tank, car

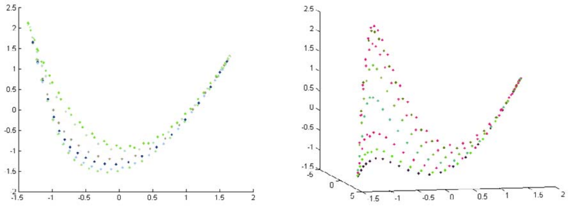
In our experiments, the training and test sets which belong to the same class both consist of 180 projection images, respectively. To acquire a balance between accuracy and efficiency, we assume the dimensionality of the embedded space to be 3. Also number of neighborhood k=6 is used in the experiments.

The process of pose estimation is to find the samples in the training set whose embedding coordinate is nearest to the one of the test samples. Obviously, the points of the same pose in the same class of the high dimensional space should be mapped into one point in the low-dimensional embedded space, and the points of the very similar pose in high dimensional space should correspond to two adjacent points in the low-dimensional embedded space.

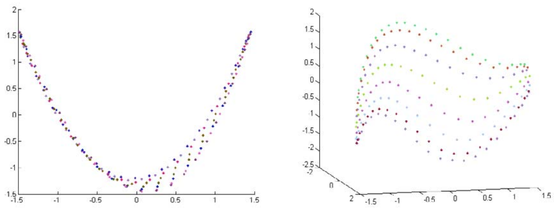
First, we compute the low-dimensional embedding coordinate of the training set of the four models and figure 4 shows the underlying 2D and 3D manifold of observational data of the corresponding four models in an embedded space using LLE. Like PCA method, in low-dimensional space of manifold, there should exist the eigen-images. Especially, for the three-dimension embedded space, there will exist three eigen-images. Using the former method, we can obtain the three eigen-images.



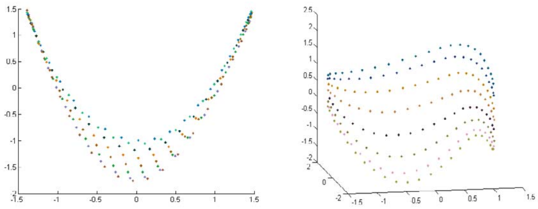
(a) Su27



(b) f16



(c) tank



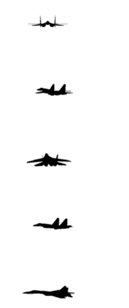
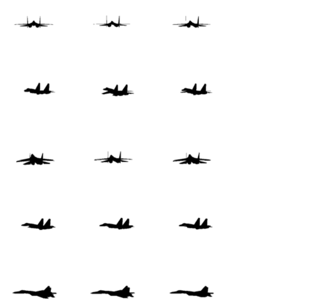
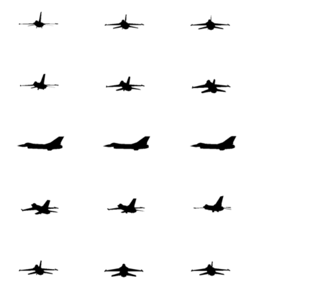
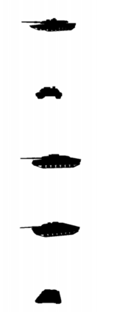
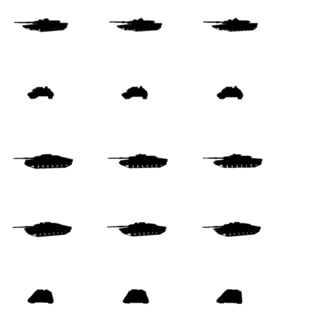
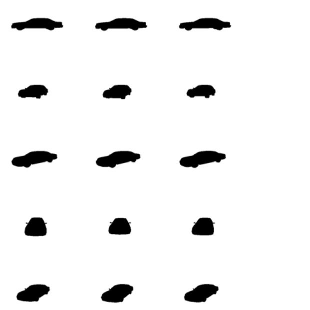
(d) car.

Figure 4. The top two (left figures) and top three (right figures) coordinate of LLE of projection images of four classes models

Next we select five poses of one model randomly (table1) from test set as input images and project them onto the space spanned by the eigen-image, then we estimate the pose by comparing its position in eigen-image with the positions of training images. The pose of the nearest position of the training images is the solution. In table 1 the first three nearest images in the training images corresponding to the input images are showed. Intuitively, the poses of the training images are very similar to the pose of the test.

Table 1 shows the input image of the four model and the corresponding first three nearest pose images in the training set resolved by using our algorithm.

Table 1. The matching images

image class	Input images	Result (first three)
Su27		
F16		
Tank		
car		

Obviously the results verify that our algorithm is feasible and is of well performance. Table 2 shows the distance from the test to the first three training images. The digits tagged red color represent the ones in table 1.

Table 2. The first three nearest distance from input to training images(C1:su27;C2:F16;C3:tank;C4:car)

C1	1st	2nd	3rd	C2	1st	2nd	3rd
1	0.030	0.031	0.031	1	0.017	0.020	0.022
2	0.013	0.031	0.049	2	0.009	0.020	0.038
3	0.046	0.063	0.092	3	0.042	0.060	0.077
4	0.006	0.014	0.019	4	0.025	0.034	0.040
5	0.031	0.041	0.044	5	0.034	0.046	0.068
6	0.077	0.086	0.088	6	0.069	0.091	0.096
7	0.019	0.021	0.022	7	0.060	0.063	0.082
8	0.027	0.083	0.104	8	0.018	0.026	0.027
9	0.063	0.069	0.076	9	0.004	0.018	0.023
C3	1st	2nd	3rd	C4	1st	2nd	3rd
1	0.028	0.053	0.060	1	0.015	0.032	0.045
2	0.053	0.093	0.100	2	0.060	0.076	0.081
3	0.044	0.071	0.074	3	0.064	0.084	0.093
4	0.010	0.028	0.045	4	0.082	0.090	0.126
5	0.093	0.100	0.102	5	0.023	0.026	0.034
6	0.044	0.045	0.053	6	0.065	0.084	0.107
7	0.010	0.011	0.014	7	0.064	0.065	0.088
8	0.042	0.115	0.118	8	0.018	0.020	0.029
9	0.016	0.022	0.025	9	0.021	0.099	0.109

In addition to, we also compare the algorithm with the method that only uses the first component of the eigen-image (see Figure 5). In figure 5, blue squares show the sequential test images on the same latitude and different longitude of view sphere, red dots denote the nearest point of our algorithm found, blue plus denote the nearest point only use the first component. Figure 5 shows that our algorithm using top three components is more robust and consistent.

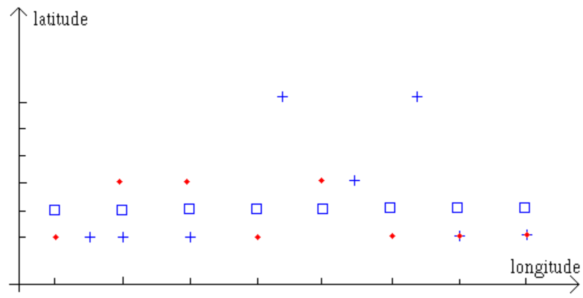


Figure 5. Comparison of matching by different component

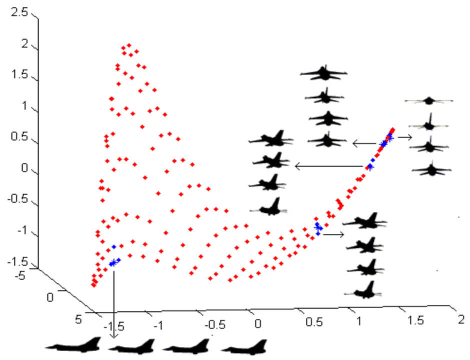


Figure 6. projection coordinate of part of the test images (F16) and their nearest training images in the embedding coordinate system

In figure 6, red dots denote the first three low-dimensional embedding coordinate of LLE of training projection-images of F16. The blue pluses denote low-dimensional projection coordinate of the test image onto the first three eigen-images computing by training images, blue dots denote the nearest low-dimensional coordinate of the test image to the training image. In figure 6 the top or the left of the five sequence images are the test images, and other three are the first three nearest training images.

5. Conclusions

In this paper, we presented an algorithm based on LLE which can estimate pose of 3D object, which can learn intrinsic geometry of the underlying manifold with metric-preserving properties and estimate the pose of the 3D object efficiently and precisely. In our algorithm we present a method to compute the eigen-images of training set by introducing the idea of PCA using the principal component deduced from LLE. And for a new sample we just compute its projection to the eigen-images. And the nearest training images from the sample are the solution. Obviously the algorithm can extend to real-time pose estimation. Experimental results show the excellent performance of our algorithm. Also the algorithm has a wide variety of potential application, especial for 3D object recognition.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (No. 60502013).

References

[1] Arbter K., SnyderW.E., Burkhardt H., Hirzinger G. "Application of affine-invariant fourier descriptors to recognition of 3-D objects". IEEE Transactions on PAMI, Vol. 12, No. 7, 1990, pp. 640－647.

[2] S. J. McKenna and S. Gong, "Real-time face pose estimation", Real-Time Imaging 4, 1998, pp. 333-347.

[3] Chibelushi, C.C.; Lund, M.L. "Robustness of head pose estimation based on Gabor wavelets and graph matching", 7th International Conference on Image Processing and Its Applications, Manchester, UK, 1999, pp. 38-41

[4] Moonhong Baeg, Hashimoto H, Harashima F, Moore J.B.; "Gradient flow approach for pose estimation of quadratic surface"[robotics control], Proceedings of the 1995 IEEE IECON, Pp. 161-166

[5] Murase H, Nayar S.K. "Visual Learning and Recognition of 3-D Object From Appearance", 1995 Int. J. Computer Vision, vol. 14, pp. 5-24.

[6] Ba S.O, Odobez J.M. "A probabilistic framework for joint head tracking and pose estimation, Pattern Recognition", 2004. Proceedings of the 17th International Conference on, Pp. 264-267

[7] Ekvall S, Hoffmann F, Kragic D. "Object recognition and pose estimation for robotic manipulation using color cooccurrence histograms", Proceedings of IROS 2003, Pp. 1284-1289

[8] Lian-wei Zhao,Si-wei Luo, Ling-zhi Liao. "3D object recognition and pose estimation using kernel PCA", Proceeding of ICMLC2004, Pp. 3258-3262

[9] Youding Zhu, Fujimura, K. "Head pose estimation for driver monitoring", Intelligent Vehicles Symposium, 2004 IEEE, pp. 501-506

[10] Zhiguang Yang, Haizhou Ai, Bo Wu, Shihong Lao. Lianhong Cai; "Face pose estimation and its application in video shot selection". Proceedings of the 17th International Conference on PR,2004, Pp. 322-325

[11] Agarwal, A., Triggs, B.: "3d human pose from silhouettes by relevance vector regression". In: CVPR. Volume II. (2004) 882-888

[12] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. Fisherfaces:Recognition using class specific linear projection", IEEE Trans. on PAMI, 19(7):711-720, 1997.

[13] M. Bartlett and T.J. Sejnowski, "Independent components of face images: A representation for face recognition", Proc. of the 4th Annual Joint Symposium on Neural Computation, 1997.

[14] J.W. Sammon, "A non-linear mapping for data structure analysis", IEEE Trans. On Computers, 18(5): 401-409, 1969.

[15] S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by locally linear embedding", Science, vol. 290, 2000.

[16] Princeton Shape Benchmark, http://shape.cs.princeton.edu/benchmark/