Robust Motion Segmentation by Clustering in Subspace

Hongbin Wang Hua Li
Institute of Computing Technology, Chinese Academy of Sciences
Beijing 100080, P. R. China
{hbwang, lihua}@ict.ac.cn

Abstract

Multibody motion segmentation is important in many computer vision tasks. One of approaches to solve this problem is the factorization approach [3]. But in practice, segmentation is difficult since the shape interaction matrix is contaminated by noise. This paper presents a novel approach to robustly segment multiple moving objects by clustering in subspace. We introduce a new affinity matrix based on the shape interaction matrix and map the feature points into a low dimensional subspace. The feature points are clustered in this subspace using a graph spectral approach. By computing the sensitivities of the larger eigenvalues of a related Markov transition matrix with respect to perturbations in affinity matrix, we improve the piecewise constant eigenvectors condition [10] dramatically. This makes clustering much reliable and robust. We confirm it by experiments.

1 Introduction

Motion segmentation is one of important tasks in computer vision. It has many applications including structure from motion, video coding and human computer interaction. Among many techniques discussed in the literature, Costeira and Kanade [3] proposed an algorithm for multibody motion segmentation based on factorization. Given tracked feature points, the technique defines a shape interaction matrix Q and groups the points into different moving clusters without 3D reconstruction. Gear [4] presented an alternative method by exploring the reduced row echelon form of measurement matrix.

The drawback of these two techniques is that the performance degrades quickly in the presence of noise. The reason is that the shape interaction matrix loses its discriminate ability when noise present. An improved approach provided by Ichimura[5,6] set threshold for Q, it suffered the same degradation. Wu[18] presented a method to separate points in subgroup level. But the subgroups are obtained using Ichimura's method by setting high threshold. In extreme case, it becomes point- by-point merging. Kanatani[8,9] proposed to work in the original data space by subspace merging, and improved the result using dimension correction and robust fitting. The subspace merging criterion and the number of objects are determined by model selection. The subspace merging technique does not guarantee the globally optimal segmentation, because it is based on local point-by-point interaction. The number of objects is critical to whole procession, but it cannot be reliably estimated by model selection.

The most related work is proposed by Inoue [7]. The absolute value of shape interaction matrix is used as affinity matrix. The feature points are mapped into a low dimensional subspace. Clusters are extracted by a graph spectral method. It has been shown that, the success of spectral clustering can guarantee by a proposition: the leading eigenvectors of a related Markov transition matrix must be roughly piecewise constant [10]. Practically, in the presence of noise, the piecewise constant eigenvectors condition breaks down. Inoue's method doesn't address this problem. So his method degrades when noise present.

In this paper, we provide a novel approach to robust segmentation of multiple moving objects by clustering in subspace. Firstly, given the number of objects, we compute shape interaction matrix in the subspace of measurement matrix. Secondly, instead of using the shape interaction matrix Q directly, we introduce a new affinity matrix based on Q. Thirdly, after mapping the feature points into a low dimensional subspace, we compute the sensitivities of the larger eigenvalues of a related Markov transition matrix when affinity matrix changes. By selecting appropriate affinity matrix and computing the sensitivity of the eigenvalues with respect to changes in affinity matrix, we improve the piecewise constant eigenvectors condition dramatically. This makes clustering procedure much reliable and well conditioned. Our approach is robust to noise due to preservation of the piecewise constant eigenvectors condition. This has been verified by extensive experiments.

The organization of this paper is as following: Section 2 defines the motion segmentation problem. Section 3 describes our approach. Section 4 provides

experimental results with both synthetic and real data. Section 6 gives conclusions.

2 Background and Basic Definitions

2.1 Shape Interaction matrix

Suppose n feature points are tracked in f frames under affine camera model. And there are N independently moving objects in the scene. The coordinate of i th point in j th frame is $\left(u_i^j, v_i^j\right)$. The coordinates of all points can be collected into a $2f \times n$ matrix

$$W = \left[\frac{X}{Y}\right]_{2f\times n} \tag{1}$$

Where

$$X = \begin{bmatrix} u_1^1 & u_2^1 & \cdots & u_p^1 \\ u_1^2 & u_2^2 & \cdots & u_p^2 \\ \vdots & \vdots & \ddots & \vdots \\ u_1^f & u_2^f & \cdots & u_p^f \end{bmatrix}_{f\times n}$$

$$Y = \begin{bmatrix} v_1^1 & v_2^1 & \cdots & v_p^1 \\ v_1^2 & v_2^2 & \cdots & v_p^2 \\ \vdots & \vdots & \ddots & \vdots \\ v_1^f & v_2^f & \cdots & v_p^f \end{bmatrix}_{f\times n}$$

According to [3], without noise and outliers, every column W lies in a 4-dimensional subspace and the rank of measurement matrix W is $4N$ where N is the number of objects. W can be decomposed by SVD

$$W = U\Sigma V^T$$

If the features from same objects are grouped together, U, Σ, V will have block diagonal structure.

$$W = \begin{bmatrix} U_1 & \cdots & U_N \end{bmatrix} \cdot \begin{bmatrix} \Sigma_1 & & \\ & \ddots & \\ & & \Sigma_N \end{bmatrix} \cdot \begin{bmatrix} V_1^T & & \\ & \ddots & \\ & & V_N^T \end{bmatrix}$$

This is because every $U_k \Sigma_k V_k^T$ is the result of single object factorization [16].

In reality, we do not know which feature belongs to which object. The feature points from different objects are mixed in the columns of W. In order to permute and group the columns of W, Costeira and Kanade [3] defined a shape interaction matrix

$$Q = VV^T \tag{2}$$

Q is motion invariant and has a property: $Q_{ij} = 0$, if point i, j belong to different objects; $Q_{ij} \neq 0$, if point i, j belong to same objects.

Kanatani [8] reformulated the property of Q as a pure mathematical theorem. Let the columns of W be n points $\{p_\alpha\}$ that belong to m linearly independent subspace $I_i, i = 1, \cdots m$, define the interaction matrix Q as (2), then

$$Q_{\alpha\beta} = 0, \qquad \alpha \in I_i, \beta \in I_j, i \neq j \tag{3}$$

This is called the subspace separation theorem. It can be proved purely mathematically without factorization.

2.2 Problem Definition

However, the rank of W is difficult to estimate even with a small noise component [7]. It makes computing the interaction matrix Q very difficult without prior knowledge of the number of objects. Even if the interaction matrix Q has been obtained, the elements of Q are nonzero in general. This makes Q lose its zero/nonzero discriminate ability.

In this paper, we want to solve the following problem: suppose points are tracked in many frames under the affine camera model, given the number of objects, how can we compute the interaction matrix Q and use Q to reliably segment feature points into multiple moving objects in the presence of noise?

3 Our Approach

3.1 Compute Q if number of objects N is known

Given the number of objects N, the shape interaction matrix can be constructed by using first $r = 4N$ column of V as (2). If N is not known, hypothesize-and-test approach [7] can be taken or N is estimated by model selection [9].

But we found that, in the presence of noise, it is very difficult to estimate the number of the objects. Firstly, we tested the model selection approach using synthetic data (see Fig.3 and experiment section). We built a

synthetic scene that consists of two sets of points. The rank of the measurement matrix is 8, but the rank estimated by model selection [11] is 3. Other test on geometric AIC and geometric MDL [9] also showed that the rank couldn't be estimated by model selection reliably.

Secondly, we look at the hypothesize-and–test approach [7]. As statement in the following section, the number of clusters is determined by the leading K eigenvectors of a matrix L that is based on Q. If L has K leading eignevectors, but the data is composed of N objects where $N > K$, the data can be grouped into K clusters at most. In this case, the hypothesize-and –test approach failed.

We observe that, if there are outliers in measurement matrix W, for example, the ith column of W, then the correspondent ith row and column of interaction matrix Q have a property: $Q_{ii} \sim 1$. $Q_{ij}, Q_{ji} \sim 0, i \neq j$. This is because the outlier only has high interaction value with itself and has low interaction with other points. This property is utilized in our approach to remove outliers.

3.2 A Basic Spectral Clustering Method

Given interaction matrix Q, it encodes pairwise interaction information of W. We propose to use this information to map the original feature points to a low dimension subspace and group the points in this subspace.

Previous work in image segmentation has implemented this idea to do bipartite graph segmentation [14,17] and extended to multipartite segmentation [10]. This can be done by casting the problem into a spectral graph clustering problem [1].

Given a $n \times n$ pairwise affinity matrix A, where A is symmetric and $a_{ij} = 0$ if point i, j belong to different clusters. Following the formulation in [2], we consider an undirected graph G with vertices $v_i, i = 1 \cdots n$, and edges $e_{ij} = a_{ij}$ which represent the affinity between vertices v_i and v_j. A Markov chain is defined by setting the transition probability $m_{ij} = d_j^{-1} a_{ij}$ where $d_j = \sum_{i=1}^{n} a_{ij}$ gives the normalizing factor which ensures $\sum_{i=1}^{n} m_{ij} = 1$. The matrix form of the definition above is:

$$M = AD^{-1}, \quad D = diag(d_1, \cdots d_n) \tag{4}$$

In practice, we consider the matrix

$$L = D^{-1/2} M D^{1/2} = D^{-1/2} A D^{-1/2} \tag{5}$$

Where L is symmetric and computationally more stable in eigen-decomposition.

Spectral clustering can be done using following simple algorithm [12]:

1. Find the leading K eigenvectors of L, if the number of clusters is known. Form the matrix $X = [V_1 \cdots V_K]$.
2. Form the matrix Y from X by normalizing each of X's rows.
3. Treating each row of Y as a point in R^K, using K-means to cluster them into K clusters.
4. Assign the original point w_i (one column in W) to clusters according to the assignment of i th row of Y.

3.3 Improving the Piecewise Constant Eigenvectors Condition

The algorithm above is only valid in ideal case. In the presence of noise, affinity matrix $a_{ij} \neq 0$ if point i, j belong to different clusters. How can we group the points into correct clusters in the noise case? It has been shown that, if the points can group into K clusters, the leading K eigenvectors of M must be roughly piecewise constant [10]. We also found that, if the leading K eigenvectors of M are roughly piecewise constant, the leading K eigenvalues of M all are 1. That is, if we can preserve the piecewise constant eigenvectors condition, the points can be grouped into several clusters without difficulty. So we propose two ways to improve the piecewise constant eigenvectors condition in the presence of noise.

The first is to choose appropriate affinity matrix. In our problem, we have got a shape interaction matrix Q where $Q_{ij} = 0$ if points i, j belong to different objects without noise. We can construct affinity matrix based on Q. One simple way is let $A = abs(Q)$ [7]. But we found that it is vulnerable to noise and easily violates the piecewise constant eigenvectors condition. We propose a new affinity matrix

$$A = \exp(-(1/abs(Q))/2\delta^2) \tag{6}$$

Where δ is a scale parameter. The Gaussian function introduces δ into affinity matrix to control the scale of interactions between points. Taking the reciprocal of the absolute value of Q should make the affinity matrix positive and $A_{ij} = 0$, if points i, j belong to different objects.

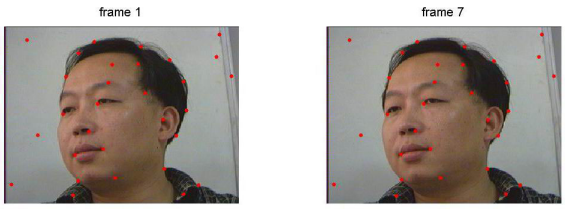

Fig. 1. Two views of a real video sequence.

We compare these two affinity matrixes using real data. We detect and track 30 feature points in 14 frames using KLT tracker [15]. Fig.1 shows first frame and 7th frame in real video data. Fig.2 shows the leading three eigenvectors of two affinity matrixes. Thought the experiments, we can see that the new affinity matrix can produce better piecewise constant eigenvectors.

The second improvement is computing the sensitivities of the larger eigenvalues of L with respect to perturbations in the edge weights. Consider symmetric matrix L, its eigen decomposition is:

$$L = U\Lambda U^T \tag{7}$$

Where $U = [\vec{u}_1, \vec{u}_2, \cdots \vec{u}_n]$ are eigenvectors. Λ is diagonal matrix composed by eigenvalues $[\lambda_1, \cdots \lambda_n]$, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$. Then the Markov transition matrix $M = D^{1/2}U\Lambda U^T D^{-1/2}$. Consider the Markov chain in graph G, it propagates t iterations. The Markov transition matrix after t iterations is:

$$M^t = D^{1/2}U\Lambda^t U^T D^{-1/2} \tag{8}$$

It can be found that M^t is completely characterized by Λ^t. In other words, the changes of L's eigenvalues reflect the changes of transition probabilities in the edges of graph G. We called this COP (Changes Of Probabilities).

For the vertices i, j belong to different clusters, the COP between them is small. In contrast, the COP within each cluster is large. This is because the connected edges between different clusters are sparse and have small weight values, and the connected edges within clusters are dense and have high weight values.

If the edge weight of a single edge between different clusters changes, the COP in this edge will more sensitive to this change because it has fewer alternative routes to take. In contrast, the COP in the edge within cluster will less sensitive to this change because it has many alternative routes to take. If we can find the edge in which the COP is more sensitive to the change of edge weight, then cut the edge. This is because, in the ideal case of well separated clusters, the weight of this edge must be zero. In the presence of noise, the well separated clusters become weakly coupled. If the linked edge generated by noise can be identified, we can cut the edge and recover the original well separated clusters.

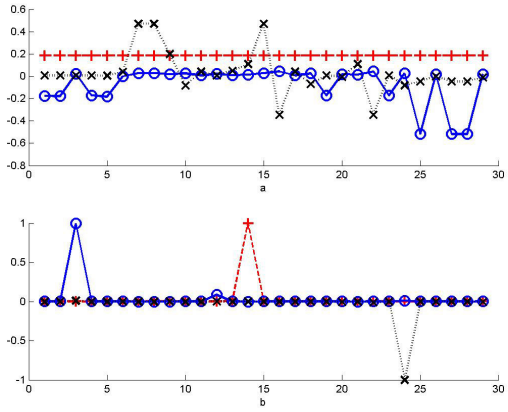

Fig. 2. The leading three eigenvectors (u_1, u_2, u_3) of two affinity matrixes. u_1 - red plus; u_2 - blue circle; u_3 - black cross. (a) eigenvectors of our new affinity matrixes. The correspondent eigenvalues is (1, 0.99949, 0.9991). (b) eigenvectors of $|Q|$. The correspondent eigenvalues is (0.045, 0.0389,0.0360).

Because the changes of L's eigenvalues reflect the COP in the edges of graph G, we compute the sensitivity of eigenvalues of L with respect to the edge weight A_{ij} [2], which represents the sensitivity of the COP with respect to edge weight.

$$\begin{aligned} S_{ij} &= \frac{d\lambda}{dA_{ij}} = \vec{u}^T \frac{dL}{dA_{ij}} \vec{u} \\ &= 2\frac{u_i u_j}{\sqrt{d_i d_j}} - \lambda\left(\frac{u_i^2}{d_i} + \frac{u_j^2}{d_j}\right) \end{aligned} \tag{9}$$

Here (u_i, u_j) are the (i, j) elements of eigenvector $\vec{u}$. (d_i, d_j) are degrees of nodes (i, j). The proof gives in the Appendix. In practice, consider larger eigenvalues of L (smaller eigenvalues have few impact on clustering), we set a threshold ε to select them except 1(The eigenvalue 1 correspond to well separated clusters and does not need to be considered). We take $\varepsilon = 0.9$.

If $|S_{ij}| > \delta \cdot median(S)$, cut the edge between i, j. δ takes a high value in order to only cut edges with the highest sensitivities.

3.4 The Algorithm

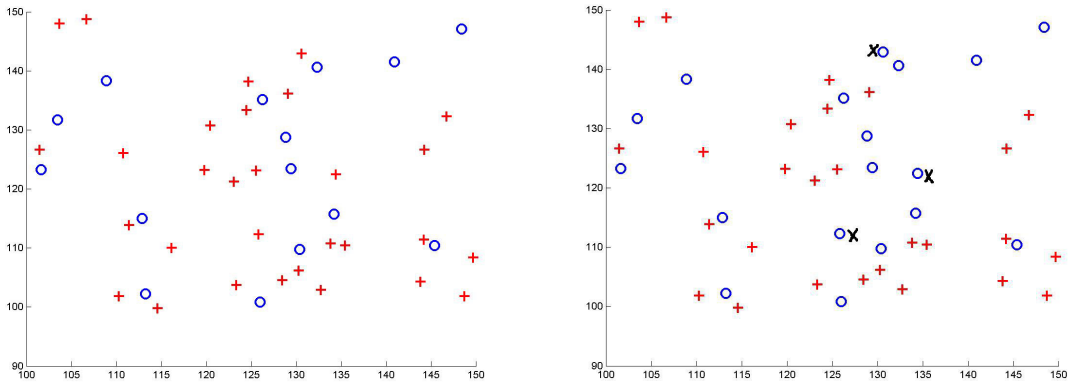

Fig.3. Segmentation result of synthetic data when noise=1.0, σ =2. There are three mis-grouping points (indicated using black cross).

Suppose n feature points are tracked in f frames under affine camera model. The coordinates of all points can be collected into a $2f \times n$ matrix W. The k th column of W represents the k th point in all frames. We want to segment the points into several clusters:

1. Given the number of objects N, decompose W by SVD: $W = U\Sigma V^T$; Compute the shape interaction matrix $Q = \hat{V}\hat{V}^T$, where $\hat{V}$ is the first $r = 4N$ columns in V.

2. If $Q_{ii} \sim 1$ and other related item $Q_{ij} \sim 0, j \neq i$, point i is outlier. Remove it from W. Return to step 1, iterate until no outlier removed.

3. Define the $n \times n$ affinity matrix A, let $A_{ij} = \exp(-(1/|Q_{ij}|)/2\sigma^2)$, σ is scale parameter.

4. Construct a diagonal matrix D, let $D_{ii} = \sum_{i=1}^{n} A_{ij}$ and form a symmetric matrix $L = D^{-1/2} A D^{-1/2}$

5. Compute eigenvectors $[V_1, V_2, \cdots V_n]$ of L, with eigenvalues $|\lambda_1| \geq |\lambda_2| \cdots \geq |\lambda_n|$.

6. Pick out larger eigenvalues $\lambda_k > \varepsilon$ except 1. Here $\varepsilon = 0.9$. Compute the sensitivities of larger eigenvalues with respect to A: $S^k = d\lambda_k / dA$.

7. If $|S_{ij}^k| > \delta \cdot median(S^k)$, set $A_{ij} = 0$. Where δ is threshold of S.

8. Recompute D, L and eigen-decomposition of L. Choose the leading K eigenvectors of L, if $\lambda_1, \cdots \lambda_K = 1$. If $K > N$, form the matrix $X = [V_1 \cdots V_N]$. Otherwise return to step 6.

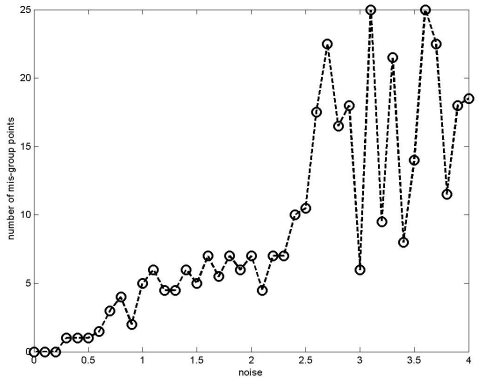

Fig. 4. Number of mis-grouping points under noise.

9. Form the matrix Y from X by normalizing each of X's rows.

10. Treating each row of Y as a point in R^N, using K-means to cluster them into N clusters.

11. Assign the original point w_i (one column in W) to clusters according to the assignment of i th row of Y.

12. Check the support of every point within each cluster according to A. Select the least support point and reassign them to clusters.

Some notes:

In the algorithm above, we have three parameters: $\sigma, \varepsilon, \delta$. The σ is the scale parameter of affinity matrix. It reflects the local scale of interaction between points. We tend to choose small σ, but if choose a very small σ, it tend to make A singular. So we choose the smallest σ while A is nonsingular. The ε controls number of larger eigenvalues. We set $\varepsilon = 0.9$. The δ is the threshold parameter to select large sensitivities in step 8. We set $\delta = 100$ to avoid over-cutting A.

K-means in the step 10 is inexpensively initialized using the prior knowledge that the clusters are about

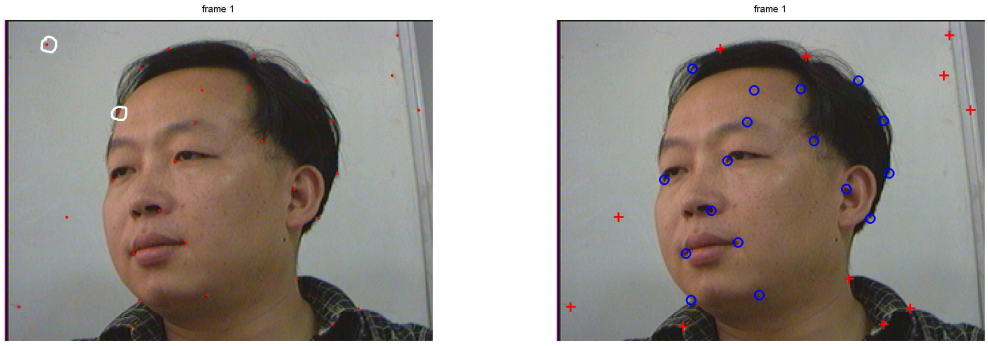

Fig. 5. Shows segmentation result in the first frame of video. Two points in white circles are regarded as outliers.

90^0 apart [12]. This makes K-means run only once to give final result and does not need to be restarted to avoid local minima.

In the last step in the algorithm, we wish to remove the outliers from final clusters. The reason is that we have found some points in the clusters often have few links and small edge weights with other points within same cluster. We must pick out these points and reassign them to clusters. If a point has comparable distances to all the clusters, it must be outlier.

The difference between our spectral clustering algorithm with eigencuts algorithm in [2] is that, our algorithm is intended to segment the points using leading eigenvectors by improving the piecewise constant eigenvectors condition in the presence of noise. The eigencuts algorithm is proposed to solve the weakly coupled data clustering problem. It chooses an eigenmode to cut edge in an iterate way. We found that it often make the matrix A singular. This is because matrix A has too many edges be cut. In contrast, our algorithm cut matrix A operates on many eigenmodes simultaneously. It is computationally more efficient and stable.

4 Experiments

In this section we provide experimental results with both synthetic and real data.

4.1 Synthetic Data

We performed some simulations to analysis our algorithm. We build a synthetic scene that consists of two sets of points. One set of 30 points placed in a 3D cube, and the other set of 15 points represents background. These two sets of points undergo different and independent motions. We generate 20 frames and add Gaussian noise to image points. Figure 3 shows the segmentation result when noise=1.0. We choose σ =2.

We also add Gaussian noise to test the robustness of our algorithm. The noise ranges from 0 to 4 with interval of 0.1. We perform 30 runs for each noise level and compute the average of the mis-grouping error. The result is showed in figure 4. We choose σ =2. It can be seen that our algorithm can give good result up to 2 unit noise. It is approximately 1.5% of image size.

4.2 Real Data

We have also applied our algorithm to some real video sequences. The first sequence contains a moving head in front of camera. We observe that, the head undergoes rotation out of plane and introduces a large perspective effect. We detect and track 30 feature points in 14 frames using KLT tracker [15] and apply our motion segmentation algorithm. Figure 5 shows our segmentation result. The two red points on the border between hair and background are grouped into background. If combined with other cues like color, our algorithm will segment these two points correctly.

We also compare three approaches: one is our approach. One is the approach presented in [12]. The last one is the approach presented in [7] which based on affinity matrix $|Q|$. Figure 6 shows the result. We can see from it that our approach performs the best.

Another sequence contains a moving hand and background. The result of segmentation is showed in Figure 7. We segment the feature points into three groups. The performance of our algorithm is excellent. But some points in the background are clustered into the same group as the moving hand. The reason is that, the KLT tracker makes the static points in background move with hand when hand passed by. The points share same motion with hand in many frames. This forces them to be clustered into same group. Segmentation of such points against the hand is very difficult.

5 Conclusions

The factorization approach to motion segmentation is based on the shape interaction matrix. However, the noise makes segmentation difficult. In this paper, we

proposed a spectral clustering approach to segment multiple moving objects robustly. By introducing a new affinity matrix and computing the sensitivities of the larger eigenvalues of L with respect to perturbations in the edge weights, we improve the piecewise constant eigenvectors condition dramatically. The feature points are mapped into a low dimensional subspace and are clustered using a spectral clustering method. The robustness of our approach is verified using synthetic and real data.

In the future, we would like to extend the work to deal with unknown number of objects case. One possible way is using EM algorithm like in [13] . We also want to apply the spectral clustering method developed in this paper to other problems like stereo correspondence.

Acknowledgements

We wish to thank Phil Culverhouse for helping the writing of the final paper. This work is supported by CAS Knowledge Creative Foundation 20016190C and Chinese 863 Foundation 2001AA231031.

Appendix

Proof of equation (9): Consider eigen decomposition of L (7), it can be written as $L\vec{u}_i = \lambda_i \vec{u}_i$, $i = 1, \cdots n$. The derivative of λ with respect to A_{ij} is:

$$S_{ij} = \frac{d\lambda}{dA_{ij}} = \vec{u}^T \frac{dL}{dA_{ij}} \vec{u}$$

Where $L = D^{-1/2} A D^{-1/2}$, so

$$\frac{dL}{dA_{ij}} = D^{-1/2}\left[Z_{ij} + Z_{ji}\right]D^{-1/2} - PAD^{-1/2} - D^{-1/2}AP$$

Where Z_{ij} is a zero matrix except item (i, j) is one. And $P = \left[\frac{d_i^{-3/2}}{2} Z_{ii} + \frac{d_j^{-3/2}}{2} Z_{jj}\right]$.

So

$$\vec{u}^T \frac{dL}{dA_{ij}} \vec{u} = \vec{u}^T D^{-1/2}\left[Z_{ij} + Z_{ji}\right]D^{-1/2}\vec{u} -$$
$$\vec{u}^T PAD^{1/2}[D^{-1/2}AD^{-1/2}]\vec{u} - \vec{u}^T[D^{-1/2}AD^{-1/2}]D^{1/2}P\vec{u}$$

Since $D^{-1/2}AD^{-1/2}\vec{u} = L\vec{u} = \lambda\vec{u}$ and $PD^{1/2} = D^{1/2}P$. So

$$\vec{u}^T \frac{dL}{dA_{ij}} \vec{u} = \vec{u}^T D^{-1/2}\left[Z_{ij} + Z_{ji}\right]D^{-1/2}\vec{u} - 2\lambda\vec{u}^T PD^{1/2}\vec{u}$$
$$= \vec{u}^T D^{-1/2}\left[Z_{ij} + Z_{ji}\right]D^{-1/2}\vec{u} - \lambda\vec{u}^T\left[d_i^{-1}Z_{ii} + d_j^{-1}Z_{jj}\right]\vec{u}$$
$$= 2\frac{u_i u_j}{\sqrt{d_i d_j}} - \lambda\left(\frac{u_i^2}{d_i} + \frac{u_j^2}{d_j}\right)$$

References

[1] Fan R. K. Chung. Spectral Graph Theory. American Mathematical Society, 1997.

[2] Chakra Chennubhotla and Allan Jepson, Half-Lives of EigenFlows for Spectral Clustering. Advances in Neural Information Processing Systems 15. MIT Press, Cambridge, MA, 2003.

[3] J. Costeira, T. Kanade. A multibody factorization method for independently moving objects. IJCV, 29(3):159-179, 1998.

[4] C. W. Gear. Multibody grouping from motion images. IJCV, 29(2): 133-150, 1998.

[5] N. Ichimura. Motion segmentation based on factorization method and discriminant criterion. Proc. 7th ICCV, pp.600-605, 1999.

[6] N. Ichimura. A robust and efficient motion segmentation based on orthogonal projection matrix of shape space. In IEEE CVPR00, pp. 446–452, South Carolina, June 2000.

[7] K. Inoue and K. Urahama, Separation of multiple objects in motion images by clustering, Proc. 8th Int. Conf. Computer.Vision, Vancouver, Canada, Vol. 1, pp. 219–224. July 2001.

[8] K. Kanatani. Motion segmentation by subspace separation and model selection. In Proc. IEEE Int'l Conf. on Computer Vision, Vancouver, Canada, July 2001.

[9] Kenichi Kanatani and Chikara Matsunaga. Estimating the Number of Independent Motions for Multibody Motion Segmentation. In ACCV02, 2002.

[10] M. Meila and J. Shi. A random walks view of spectral segmentation. Proc. International Workshop on AI and Statistics, 2001.

[11] Thomas P. Minka, Automatic choice of dimensionality for PCA. Advances in Neural Information Processing Systems 13. MIT Press, Cambridge, MA, 2001.

[12] A. Ng, M. Jordan and Y. Weiss, On Spectral Clustering: analysis and an algorithm. Advances in Neural Information Processing Systems 14. MIT Press, Cambridge, MA, 2002.

[13] A. Robles-Kelly, E. R. Hancock, Pairwise Clustering with Matrix Factorisation and the EM Algorithm. In Proc. ECCV02, 2002.

[14] J. Shi and J. Malik. Normalized cuts and image segmentation. IEEE Transaction on Pattern Analysis and Machine Intelligence, 2000.

[15] J. Shi and C. Tomasi. Good feature to track. In IEEE Conf. on Computer Vision and Pattern Recognition, pages 593-600, Seattle, June 1994.
[16] N. Tishby and N. Slonim. Data clustering by Markovian Relaxation and the Information Bottleneck Method. NIPS, v 13, MIT Press, 2001.
[17] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. IJCV, 9(2): 137-154, 1992.
[18] Y. Weiss. Segmentation using eigenvectors: a unifying view. Proc. 7th ICCV, pp.975-982, 1999.
[19] Ying Wu, Zhengyou Zhang, Thomas S. Huang, John Y. Lin. Multibody Grouping via Orthogonal Subspace Decomposition, In IEEE CVPR01, 2001.

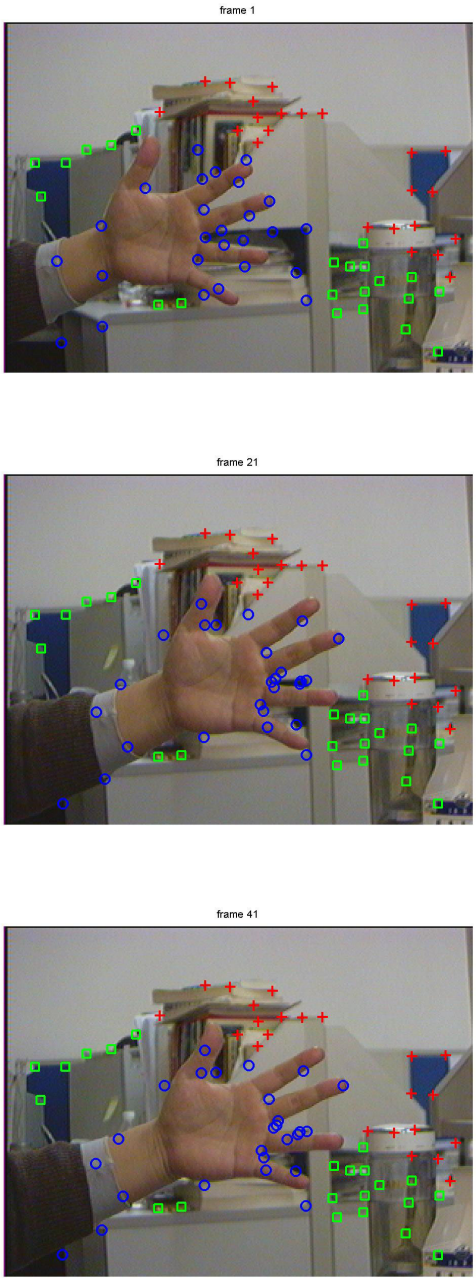

Fig. 7. Segmentation result on a hand sequence.

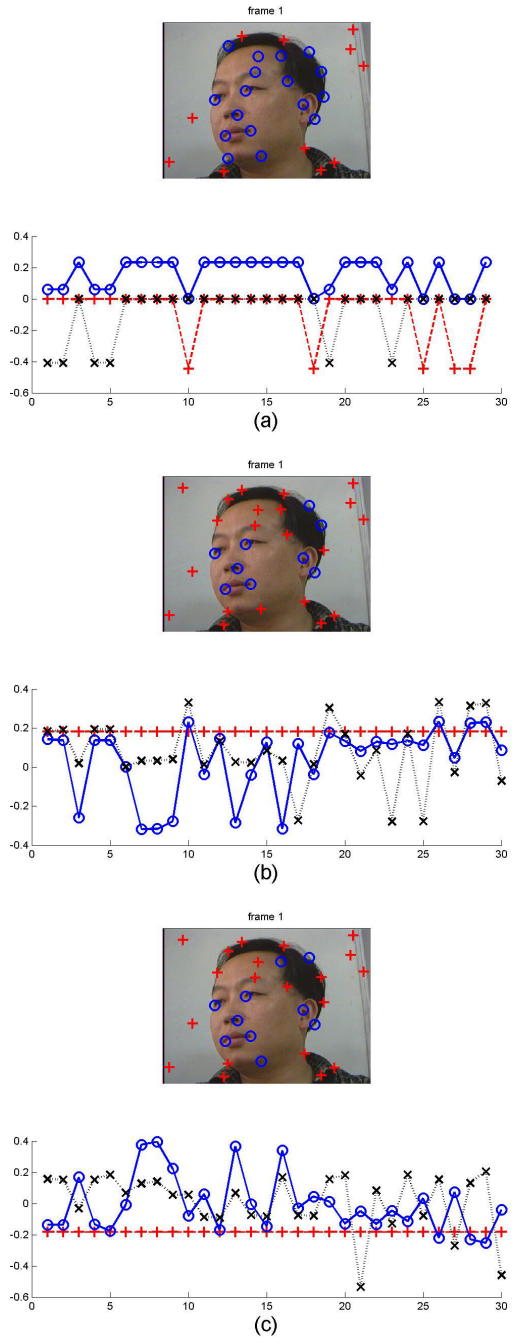

Fig. 6. Segmentation results and the leading three eigenvectors (u_1, u_2, u_3) of L. u_1 - red plus; u_2 - blue circle; u_3 - black cross. (a) our segmentation result. The correspondent eigenvalues is (1, 1, 1). (b) Segmentation result based on algorithm in [12] without computing the larger eigenvalues of L with respect to perturbations in the edge weights. The correspondent eigenvalues is (1, 0.98419 , 0.97602). (c) Segmentation result based on affinity matrix $|Q|$. The correspondent eigenvalues is (1, 0.48654, 0.43743).