Determining Correspondences and Rigid Motion of 3-D Point Sets with Missing Data *

Xiaoguang Wang, Yong-Qing Cheng, Robert T. Collins, and Allen R. Hanson

Department of Computer Science
University of Massachusetts, Amherst, MA. 01003-4610
Email: {xwang, cheng, rcollins, hanson}@cs.umass.edu

Abstract

This paper addresses the general 3-D rigid motion problem, where the point correspondences and the motion parameters between two sets of 3-D points are to be recovered. The existence of missing points in the two sets is the most difficult problem. We first show a mathematical symmetry in the solutions of rotation parameters and point correspondences. A closed-form solution based on the correlation matrix eigenstructure decomposition is proposed for correspondence recovery with no missing points. Using a heuristic measure of point pair affinity derived from the eigenstructure, a weighted bipartite matching algorithm is developed to determine the correspondences in general cases where missing points occur. The use of the affinity heuristic also leads to a fast outlier removal algorithm, which can be run iteratively to refine the correspondence recovery. Simulation results and experiments on real images are shown in both ideal and general cases.

1 Introduction

Given a set of 3-D points on a rigid body in one Cartesian coordinate system and another set of 3-D points from the same body in a rotated and translated coordinate system, we seek to estimate the rotation and translation parameters while recovering correspondences of the points in the two sets. The difficulty of this problem is due to a cyclic dilemma: it would be easy to solve for the motion parameters if we knew the correspondences of the 3-D points at two time instances, and the correspondences would not be hard to obtain if the motion parameters were known; but when neither the correspondences nor the motion parameters are known, simultaneously solving for both becomes difficult.

Previous motion research has been divided roughly into two approaches, depending on whether or not the correspondences are recovered. Some researchers (e.g. [2]) try to recover the point correspondences using combinatorial searches. These methods are very time-consuming. The other research trend is the estimation of motion parameters without knowing correspondences [1, 4, 5, 7, 8, 9]. However, correspondenceless algorithms work in a very narrow domain, called the *ideal motion problem*, where the numbers of points in the two sets are equal and each point in a set must have a corresponding point in the other set. Correspondenceless algorithms for *general motion problems*, where missing data (or *outliers*) occur, have not been reported in the literature.

The philosophical basis of this paper is that we believe correspondence recovery and refinement are necessary in dealing with data in real world where noise and outliers are inevitable. The methodological question follows: without knowing the motion parameters, is it possible to recover and refine the correspondences efficiently? We give an affirmitive answer to the question with a new method for solving the correspondence problem. The method is based on the observation that the correlation matrix of the observed 3-D point set has the property of eigenstructure invariance with respect to point permutation. With this property, a closed-form solution can be obtained for ideal motion and correspondence problems. The immediate goal of the new method is to recover a permutation matrix representing the correspondences of the points. In general motion problems, where outliers exist, the permutation matrix recovery is replaced by a weighted bipartite matching algorithm, which has a solid foundation in graph theory. Badly-matched points are detected and removed as outliers. As a result, the method is able to deal with general motion problems. In summary, we break up the chicken-and-egg dilemma of resolving motion and correspondences by recovering the correspondences first, from which we try to determine the "good" ones and use them to estimate the motion. This is different from the global approximation philosophy of correspondenceless methods.

2 Ideal Case: a Closed-Form Solution

In this paper, motion is defined in terms of a fixed 3-D point $\mathcal{S}_i$ represented in two $\mathbf{R}^3$ Cartesian coordinate systems $\mathcal{O}_A$ and $\mathcal{O}_B$. Suppose $\mathcal{S}_i$'s coordinates are observed to be a_i and a_i' in $\mathcal{O}_A$ and $\mathcal{O}_B$, respectively. a_i and a_i' are related by

$$a_i' = Ra_i + T, \tag{1}$$

in which R is a 3×3 *orthogonal matrix* that reflects the rotation between $\mathcal{O}_A$ and $\mathcal{O}_B$, and T a 3×1 vector reflecting the translation between $\mathcal{O}_A$ and $\mathcal{O}_B$. We

*This work was funded under ARPA/Army TEC contract DACA76-92-C-0041 and NSF grant CDA-8922572.

are given two sets of observed points in $\mathcal{O}_A$ and $\mathcal{O}_B$: $\mathbf{A} = \{a_i\}_{i=1}^m$ and $\mathbf{B} = \{b_j\}_{j=1}^n$, but we don't know the correspondence of any particular $a_i \in \mathbf{A}$ and $b_j \in \mathbf{B}$. The *motion problem* is to estimate R and T from these observed points. The *correspondence problem* is to recover the correspondences of the points in $\mathbf{A}$ and $\mathbf{B}$. A general motion and correspondence problem may have data with $m \neq n$, signifying non-corresponding points in $\mathbf{A}$ and $\mathbf{B}$. In this section we first investigate the ideal case where $m = n$ and assume that, for each $a_i \in \mathbf{A}$, there always exists an $a_i' \in \mathbf{B}$ satisfying (1), and vice versa.

Juxtaposing the observed vectors, we get two *observation matrices*: $A_{3\times m} = [a_1, ..., a_m]$ and $B_{3\times n} = [b_1, ..., b_n]$. Each column a_i of A may have its corresponding point residing at b_j in B, i.e. $b_j = a_i' \in \mathbf{B}$. A good property of the ideal case is that the geometric centers of $\mathbf{A}$ and $\mathbf{B}$ satisfy equation (1) as well. This property eliminates T's influence when we estimate R. Formally, let $c_A = \sum_{i=1}^m a_i/m$ and $c_B = \sum_{j=1}^n b_j/n$. In a manner parallel to the observation matrices, we define the *centralized observation matrices* by $A_C = [\alpha_1, ..., \alpha_m]$ and $B_C = [\beta_1, ..., \beta_n]$, where $\alpha_i = a_i - c_A$ and $\beta_j = b_j - c_B$. It holds that, if $a_i \in \mathbf{A}$ and $a_i' \in \mathbf{B}$ satisfy (1), then

$$\alpha_i' = R\alpha_i, \tag{2}$$

where $\alpha_i' = a_i' - c_B$ is α_i's corresponding point in $\mathbf{B}$.

2.1 Permutation Matrix

Now we introduce a *permutation matrix*, P, as a tool for solving the correspondence problem. Given an arbitrary matrix Z, matrix P is a permutation matrix iff the right-hand-side multiplication $Z_P = ZP$ is just a re-ordering of the columns of Z. Given a matrix Z and its column re-ordered version Z_P, there may exist many permutation matrices, among which there always exists one in *strict form*. (A strict form of permutation matrices is an identity matrix with its rows re-ordered [11].) With this property and equation (2), the following claim holds: in the ideal motion problem, there always exists an $m \times m$ permutation matrix P such that

$$B_C = RA_CP; \tag{3}$$

particularly, there is such a P in strict form. Equation (3) shows that the correspondence problem of the centralized observation matrices can be described by a permutation matrix.

2.2 Eigenstructure Decomposition

We define *correlation matrices* of the centralized observation matrices as $A_C^TA_C$ and $B_C^TB_C$, which are symmetric matrices. From equation (3), we have

$$B_C^TB_C = P^TA_C^TR^TRA_CP = P^TA_C^TA_CP \tag{4}$$

in which R is eliminated due to its orthogonality. When P is orthogonal, equation (4) is a similarity transformation, and the eigenstructure of $A_C^TA_C$ and $B_C^TB_C$ are invariant. From eigenstructure decomposition theory, there exist orthogonal matrices U_A and U_B and a diagonal matrix D, such that $A_C^TA_C = U_ADU_A^T$ and $B_C^TB_C = U_BDU_B^T$. $A_C^TA_C$ and $B_C^TB_C$ share the same *eigenvalues* on D's diagonal. The columns of U_A and U_B are known as *eigenvectors*. One solution of P is given by

$$P = U_AU_B^T. \tag{5}$$

This tells us that the correspondence permutation matrix can be obtained in a closed form from a very simple computation. The matrices D, U_A and U_B can be calculated from standard eigenstructure decomposition procedures. (Equation (5) holds only if the signs of the eigenvectors in U_A and U_B are consistent.)

It is worth noting that there is a symmetry [11] between our method and the traditional correspondenceless methods [1, 4, 5, 7, 8, 9]. Based on an eigenstructure decomposition of the 3×3 *scatter matrices* $A_CA_C^T$ and $B_CB_C^T$, correspondenceless methods use the orthogonality property of P to solve for R, while our method uses the the orthogonality of R to solve for P, as seen in equation (4). The symmetry reflects the inherent cyclic dilemma of motion and correspondence problems and the different strategies to solve them. Further investigations of this eigenstructure symmetry lead to some properties that are useful in implementations of the new method. First, $A_C^TA_C$ and $B_C^TB_C$ have as many nonzero eigenvalues as the scatter matrices do. Hence, the complete eigenstructure of a correlation matrix can be represented using only three eigenvectors. Second, the eigenvectors can be found efficiently. Third, one solution of P can be obtained in the following simplified representation. Let $W = \text{diag}(d_1, d_2, d_3)$, $Q_A = [u_1, u_2, u_3]_{m\times 3}$, and $Q_B = [v_1, v_2, v_3]_{n\times 3}$, in which d_1, d_2, d_3 are the three nonzero eigenvalues, and u_1, u_2, u_3 and v_1, v_2, v_3 are the associated eigenvectors of $A_C^TA_C$ and $B_C^TB_C$, respectively. We then have the permutation matrix (generally not in strict form)

$$P = Q_AQ_B^T \tag{6}$$

as a solution of equation (3). Fourth, the eigenvectors' sign issue is restricted to Q_A and Q_B, and is equivalent to that of scatter matrices (See [1, 7, 8]). Finally, define the column vectors of Q_A^T and Q_B^T as *feature vectors*. It holds that the column correspondences of A_C and B_C are equivalent to feature vector correspondences. In fact, in ideal cases without noise, each feature vector in Q_B^T has an identical feature vector in Q_A^T, and vice versa.

2.3 Solving for R and T

Recall equation (3). After we have solved for P, A_CP can be obtained by permutating the centralized observation matrix A_C. Each column of A_CP is a corresponding vector to the same column in B_C. Adding back the center vectors c_B and c_A to the columns of B_C and A_CP, we will get the the original observation matrix B and a permutated matrix A' of A. B and A' are correponding matrices, reflecting the correspondences in $\mathbf{A}$ and $\mathbf{B}$.

So far we have solved the ideal 3-D correspondence problem in a closed form, and have reduced the original problem to estimation of rotation and translation from 3-D points with known correspondences. Several noise-resistant solutions to this problem have been published (e.g. [12]). One of the advantages of our method over correspondenceless methods is that it can make use of these noise-resistant algorithms directly, while correspondenceless methods cannot.

3 General Correspondence Recovey

A shortcoming of the closed-form solution is that it is limited to ideal motion problems. In general cases the two sets of observed points are obtained separately from different observing environments, such that there will usually be some points in **B** that have no corresponding points in **A**, and vice versa. Therefore, equation (3) cannot be established, and P cannot be determined by (6). In this section, we extend the correlation matrix decomposition method to deal with general motion problems, that is, we try to determine the corresponding point pairs from the two observed point sets while removing the non-corresponding points.

Eigenstructure-based algorithms depend heavily on the correctness of the eigenstructure of the observed data. Outliers can ruin the eigenstructure, especially when they severely bias the geometrical centers of the observed data sets. In this paper, we deal with the case where the outliers are not so dominant and the observed point sets still "look like" a rigid body.

3.1 Affinity

An *affinity* is a measurement of a pairing likelihood of two objects in two sets. Our task is to establish a pairing affinity of the observed points in **A** and **B**. Pairs with high affinity values are to be considered the corresponding points; others to be outliers.

From Section 2 we know that the correspondences of **A** and **B** can be represented by feature vector correspondences in Q_A^T and Q_B^T. We define the pairing affinity of the observed points as eigenvalue-weighted, negative Euclidean norms of the feature vectors' distances in their eigenvector matrices. Formally, let d_{Ak} and d_{Bk} denote the nonzero eigenvalues of $A_C^T A_C$ and $B_C^T B_C$, and $u_k = [p_{1k}, ..., p_{mk}]^T$ and $v_k = [q_{1k}, ..., q_{nk}]^T$ the corresponding eigenvectors, $k = 1, 2, 3$. (Thus $p_i = [p_{i1}, p_{i2}, p_{i3}]^T$ and $q_j = [q_{j1}, q_{j2}, q_{j3}]^T$ are feature vectors.) The affinity of p_i and q_j is defined by

$$h_{ij} = - \sum_{k=1,2,3} d_{Ak} d_{Bk} \parallel p_{ik} - q_{jk} \parallel^2 . \qquad (7)$$

h_{ij} is an increasing function with respect to the similarity of the two feature vectors. The multiplications by eigenvalues make the eigenvectors that correspond to bigger eigenvalues have more influence on h_{ij}. Some authors [10] use similar affinity definitions in solving correspondence problems. But motion issues and outlier removal issues are not their main focus.

3.2 Weighted Bipartite Matching

Weighted bipartite matching is a mathematical tool which can be used to solve motion correspondence problems with affinity. Let $G(V_A, V_B, E)$ be an undirected complete *bipartite graph*, whose nodes are partitioned into two disjoint sets V_A and V_B. The edge set E consists of all the edges between nodes in V_A and V_B. A subset $M \subseteq E$ is said to be a *matching* if no two edges in M are incident on the same node. Given an edge-weighted bipartite graph, a *weighted bipartite matching problem* (also known as *assignment problem*) is to find a matching for which the sum of the weights of the edges is maximum.

The correspondence problem with the affinity measure defined above can be modeled by an assignment problem. The partitioned node sets V_A and V_B are composed of the feature vectors in the eigenstructure matrices Q_A and Q_B. The weight associated with each edge is the affinity h_{ij}. A matching algorithm will find a maximum weighted matching such that the feature vectors have the most affinity to each other between Q_A and Q_B.

An efficient weighted bipartite matching algorithm with expected time complexity $O(mn \log(\min(m, n)))$ has been proposed by Karp [6], provided that the weights of the edges from any fixed node in V_A are identically distributed random variables. We adopt this algorithm to solve for the correspondences of the feature vectors. The identical distribution requirement is satisfied in that we don't confine our rigid body to certain sizes and shapes.

3.3 Outlier Removal

The weighted bipartite matching algorithm results in a matching that describes the point correspondences in **A** and **B**. Although Least Median Square (LMS) methods are applicable to the matched edges for detecting outliers, they are time-consuming. Here we propose a fast outlier removal algorithm, which makes further use of the affinity heuristic.

The strategy we take in the algorithm is conservative: we aim to remove as many incorrect correspondences as possible, rather than to keep as many correct correspondences as possible. First, the matching algorithm always outputs a matching with cardinality $\min(m, n)$. Suppose $m < n$. We remove as outliers all the $(n-m)$ nodes in V_B that are not matched with any nodes in V_A, because they have low affinities with V_A. Second, those edges with large affinity values tend to be true correspondences, and those with small affinity values tend to be outliers. A partitioning operation is applied to the matching $M = M' + M''$, in which any edge in M' has a higher weight than those in M''. The set M'' is then removed as outliers. The heuristic criterion we use for partitioning M is to minimize the function

$$t(M', M'') = |s(M') - \gamma s(M'')|, \qquad (8)$$

in which $s(M')$ and $s(M'')$ are the sums of the weights of the edges in M' and M'', and γ is an empirical parameter that affects the size of M'' to be removed.

Due to errors in eigenstructures, the affinities obtained from Q_A and Q_B are erroneous. Some true correspondences may be associated with low affinity values, and thus removed as a result of the above two steps. It's easy to understand that for motion estimations keeping true correspondences is less important than removing false ones. Here we sacrifice some true correpondences for a faster outlier removal.

The ability of outlier removal is one of the most distinguishing properties of the correlation matrix based eigenstructure decomposition method. The motion parameters then be estimated from the "purified" set M'. When outliers exist, correspondenceless algorithms have a hard time determining accurate motion estimates because they never perform a correspondence analysis. Therefore, the new method has a much wider application domain.

3.4 Algorithm Outline

The following is an outline of the outlier-removing 3-D rigid body motion estimation algorithm, with observation matrices A and B as input, and motion parameters R and T as output.

1. Centralize A and B and compute the correlation matrices $A_C^T A_C$ and $B_C^T B_C$.

2. Apply eigenstructure decomposition to $A_C^T A_C$ and $B_C^T B_C$ to get the eigenvalues and the eigenvector matrices Q_A and Q_B.

3. Compute the affinity h_{ij}'s using formula (7).

4. Compute the maximum weighted matching M using Karp's algorithm.

5. Get M' by outlier removal from M.

6. Compute R and T by applying noise-resistant algorithms (e.g. [12]) to the correspondences in M'.

Some implementation issues regarding the algorithm, such as the eigenvector sign issue, are discussed elsewhere [11]. We point out that better estimates of the motion parameters can be obtained by iteratively running the algorithm. Some outliers are removed in each iteration, resulting in better estimates of the geometric centers in the next iteration, in which the matching can be refined gradually.

4 Simulations

Simulations are performed on a set of 3-D points representing a rigid body. 20 points are randomly generated with a uniform distribution in a $(100\times100\times100)$ volume. The points are then transformed to a rotated and translated coordinate system as the second observation data set of the rigid body. The translation vector is set to $(10, 20, 30)$. The rotation is set to $(40°, 50°, 60°)$ in Euler Angles. The points in the two sets are then reordered randomly. The experiments are designed to test the effects of both noise and outliers on algorithm performance. Gaussian noise, with zero mean and variance σ ranging from 0 to 9, is added to each point in the two sets. To simulate outliers, we use a random dropping technique to drop off l_a points from one set and l_b points from the other. No corresponding point pair is dropped from the two sets. That is, the input sets of the algorithm are $\mathbf{A}$, with $(20-l_a)$ points, and $\mathbf{B}$, with $(20-l_b)$ points, and there are $(20-l_a-l_b)$ pairs of corresponding points and l_a and l_b outliers in $\mathbf{A}$ and $\mathbf{B}$, respectively.

We propose a *hit rate*, ρ, as the evaluation for the performance of the algorithm:

$$\rho = |M_{hit}|/|M|, \tag{9}$$

in which M is the output matching of the algorithm, and $M_{hit} \subseteq M$ is the subset of correct correspondences in M. Two iterations are executed for the hit rate analysis. The first iteration is to get the initial matching M_0 of $\mathbf{A}$ and $\mathbf{B}$. The second is to get a new matching M_1 out of M_0', which is the purified matching using criterion (8). We set $\gamma = 1$ in the criterion. For each configuration of (σ, l_a, l_b), the algorithm is performed one thousand times, from which we get $\bar{\rho}_0$ and $\bar{\rho}_1$ (the average hit rates of M_0 and M_1), and sd_0 and sd_1 (their standard deviations).

Our first experiment is on perfectly corresponding sets: there are no missing points in $\mathbf{A}$ and $\mathbf{B}$. The curves of average hit rates as a function of noise variance are shown in Figure 1. The second experiment is to test the algorithm's performance on outliers without noise (Figure 2). We keep $l_b = 0$ but drop points from $\mathbf{A}$. So there are l_a points in $\mathbf{B}$ missing their correspondences in $\mathbf{A}$. The third experiment is on testing noisy data with outliers. The hit rate response curves for this condition are shown in Figures 3 and 4 for two particular cases: 1) $l_a = l_b = 2$, and 2) $l_a = 4, l_b = 0$.

Some conclusions can be drawn from the simulation results. Firstly, the algorithm performs well in the presence of noise. For ideal motion problems where there are no ouliers (Figure 1), the correspondences are perfectly recovered when the noise is small. When noise increases, the stability goes down only slowly. For general cases (Figures 3 and 4), the average hit rate curves show the same flatness. Secondly, when outliers exist, the outlier removal iteration makes a significant improvement in performance, with or without noise. From Figures 2, 3 and 4 we can see that the average hit rate rises about 10 percentage points consistently at any number of outliers and any noise levels. Thirdly, when the number of outliers increases, the hit rate drops. However, with the improvement by the outlier removal iteration, correct correspondences (on average) account for the majority in the resulting matchings over a wide range of input conditions. In Figure 2, when nearly half of $\mathbf{B}$ miss their corresponding points in $\mathbf{A}$, the algorithm still results in matchings with about 50% correct correspondences on average. In those cases, the LMS method could be applied to the resulting matchings for more refinement.

5 An Experiment on Real Images

The 3-D correspondence algorithm is applied to a real image sequence. The sequence was captured by a SONY B/W AVC-D1 camera fixed to a PUMA arm, which was moving in a room. The camera has an effective field of view of approximately $40° \times 40°$. Its intrinsic parameters were calculated from the manufacturer's specification sheets. Relative poses between frames were recorded when capturing the pictures. The images were digitized to 256×256 pixels.

Two sources of 3-D data are used in the experiment. One source, refered to as *model*, are points measured

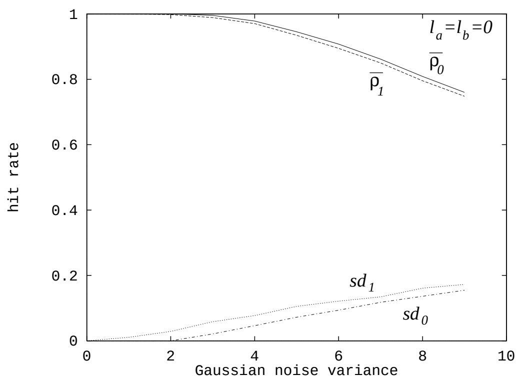

Figure 1: Performance against noise without outliers

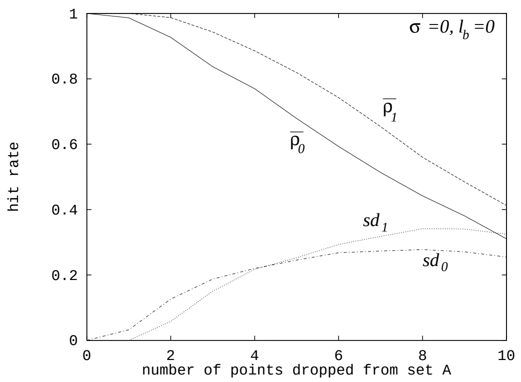

Figure 2: Performance against outliers without noise

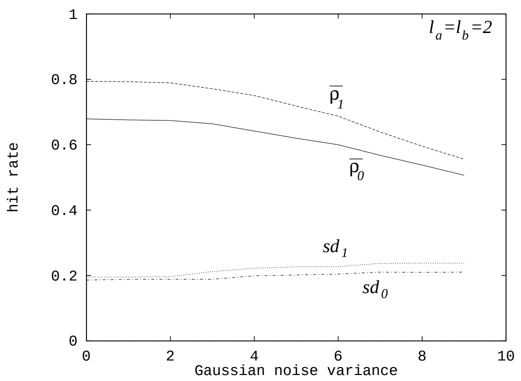

Figure 3: Performance against noise with outliers

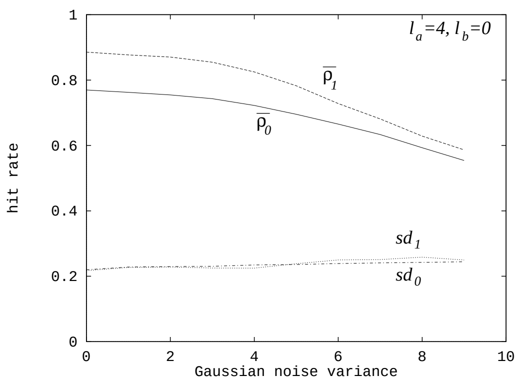

Figure 4: Performance against noise with outliers

on the objects in advance with respect to a world coordinate system. 30 model points were chosen and measured to an accuracy of approximately 0.2 feet along each axis. They are shown in Figure 5(Left), overlaid on the first image of the sequence. The depths of these model points from the camera vary from 13 feet to 33 feet in the sequence. The other data source is *triangulated data*, which are acquired by a triangulation procedure using corresponding 2-D image data [3], and represented in a local camera coordinate system. As seen in Figure 5(Middle), some model points were not captured in the image sequence due to the limited view of the camera. Only 26 points were successfully triangulated. The eigenvalues of the correlation matrices of the model and the triangulated data are (1351, 322, 77) and (1084, 254, 63), respectively.

A "model matching" experiment is designed for the image sequence. The goal is to determine the transformation between the local camera coordinate system and the world coordinate system through matching the triangulated data with the model. This is a general motion problem as missing data occur.

Both model and triangulated data are noisy, due to errors in measurement and in low-level image processing. To estimate the noise between two 3-D point sets, we transform the triangulated data to the model's coordinate system and calculate the deviation of each corresponding point. (Note: this is just for estimating the level of noise in our data; the transformation between the two sets are typically not available beforehand in reality.) The average and maximal deviation are 0.30 feet and 1.25 feet. For comparison, we calculate for each model point the distance to its nearest point (called DNP). The average and minimal DNP's of the model are 1.46 feet and 0.65 feet. This tells us that the triangulated data are very noisy, as the maximal error in the triangulated data are even greater than the distance of a pair of points in the model.

The algorithm starts with the 30 and 26 points in the two data sets, and gets an initial matching of 26 points in both sets, among which there are 16 correct corresondences. We run the fast outlier removal algorithm iteratively with $\gamma = 1$. The algorithm ends up with a fully correct matching after 7 iterations. The variation of the hit rates in these iterations are shown in Figure 5(Right). At the last iteration, the cardinality of the matching is 11. That is, we find a correct matching of 11 points in both sets without the help of LMS methods. Using this correct matching, the motion between the local camera coordinate system and the world coordinate system is recovered and shown in Table 1. It is seen that the recovered parameters are very accurate, indicating that the algorithm removes in its iterations not only the outliers but also some corresponding point pairs whose positions are severely corrupted by noise.

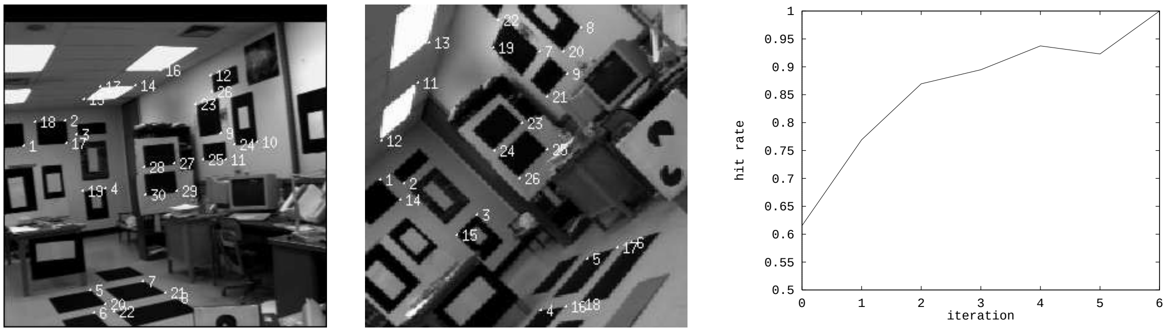

Figure 5: Model matching experiment (Left: a measured 3-D model overlaid on an image; Middle: triangulated 3-D points in a local camera coordinate; Right: the hit rate in each iteration of the outlier removal algorithm)

Table 1: The actual and recovered motion parameters between the model and the triangulated data

	Actual Parameters	Recovered Parameters
θ	$-163.37°$	$-163.52°$
ϕ	$17.40°$	$17.73°$
ψ	$-131.80°$	$-131.74°$
T_x	0.05 ft	0.11 ft
T_y	-3.11 ft	-3.07 ft
T_z	32.57 ft	32.57 ft

6 Conclusions and Discussions

We emphasize the importance of correspondence recovery and outlier elimination in 3-D motion problems, and adopt the strategy of determining correspondences before estimating motion parameters. Disclosing a theoretical symmetry in motion and correspondence problems, we propose a closed-form solution to ideal motion problems based on correlation matrix eigenstructure decomposition, parallel to the traditional scatter matrix eigenstructure decomposition. Using a heuristic measure of point pair affinity computed from the eigenstructure, a weighted bipartite matching algorithm is adopted to determine the correspondences in general motion problems. By using the affinity heuristic again, a fast outlier removal algorithm is proposed to improve hit rates in bipartite matchings. Simulations and experiments on real images show that the algorithms perform well to refine the correspondences by rmoving outliers and badly corrupted data, and potentially have a wider application domain than traditional methods.

The philosophy and methods developed in this paper are not confined to 3-D rigid motion. They could be extended to 2-D and/or deformable motion problems. They could also be extended to correspondence problems in other feature space such as colors. Further issues to consider include studying the limitations of the eigenstructures-based methods. When too many outliers exist in the point sets, the eigenstructure is totally destroyed and such methods are problematic. It seems that knowledge-based pre-processing procedures are possible in many circumstances to pre-process the observed data and make them applicable to the algorithms developed in this paper.

References

[1] J. (Y.) Aloimonos and I. Rigoutsos, "Determining the 3-D motion of a rigid surface patch without correspondence under perspective projection," *Proc. AAAI*, pp. 681-688, August 1986.

[2] J.-C. Cheng and H.-S. Don, "A structural approach to finding the point correspondences between two frames," *Proc. IEEE Int. Conf. on Robotics and Automation*, vol. 3, pp. 1810-1815, April 1988.

[3] Y.-Q. Cheng, R. Collins, A. Hanson and E. Riseman, "Triangulation without correspondences," *Proc. Arpa Image Understanding Workshop*, pp. 993-1000, 1994.

[4] D. B. Goldgof, H. Lee and T. S. Huang, "Matching and motion estimation of three-dimensional point and line sets using eigenstructure without correspondences," *Pattern Recognition*, vol. 25(3), pp. 271-286, 1992.

[5] E. Ito and J. (Y.) Aloimonos, "Is correspondence necessary for the perception of structure from motion?" *Image Understanding Workshop*, pp. 921-929, 1988.

[6] R. M. Karp, "An algorithm to solve the $m \times n$ assignment problem in expected time $O(mn \log n)$," *Networks*, vol. 10, pp. 143-152, 1980.

[7] C.-H. Lee and A. Joshi, "Correspondence problem in image sequence analysis," *Pattern Recognition*, vol. 26, pp. 47-61, 1993.

[8] Z.-C. Lin, H. Lee and T. S. Huang, "Finding 3-D point correspondences in motion estimation," *Proc. Eight Int. Conf. on Pattern Recognition*, pp. 303-305, 1986.

[9] S.-C. Pei and L. G. Liou, "Using moments to acquire the motion parameters of a deformable object without correspondences," *Image and Vision Computing*, vol. 12, no. 8, pp. 475-485, October 1994.

[10] L. S. Shapiro and J. M. Brady, "Feature-based correspondence: an eigenvector approach," *Image and Vision Computing*, vol. 10(5), pp. 283-288, June 1992.

[11] X. Wang, Y.-Q. Cheng, R. T. Collins, A. R. Hanson, R. Weiss and R. Moll, "Using weighted bipartite matching to solve the general 3-D rigid motion problem," Technical Report 95-106, Dept. of Computer Science, Univ. of Massachusetts, 1995.

[12] Z. Wang and A. Jepson, "A new closed-form solution for absolute orientation," *Proc. IEEE Conf. Computer Vision Pattern Recognition*, pp. 129-134, 1994.