Proc. First Conference on A.I. Applications, Dec. 1984

ESTIMATING 3-D MOTION FROM RANGE DATA

S.D. Blostein and T.S. Huang

Department of Electrical Engineering and Coordinated Sciences Laboratory
1101 W.Springfield Ave., University of Illinois, Urbana, Illinois 61801

Abstract

Methods are developed for estimating 3-D general motion (rotation and translation) parameters from 3-D range data at two time instances. The problem can be expressed in terms of linear equations consisting of correspondences of either points or unit direction vectors joining these points. Alternatively, no equations need be solved and nonlinear expressions be evaluated, as, for example, in the case of the screw motion method to be described. It will be shown that the best method (in terms of numerical accuracy) involves first obtaining two unit direction vectors from three correspondences, constructing a third orthogonal direction vector, and solving a three by three linear system to obtain the rotation. The translation then follows trivially once the rotation is determined. Simulation results are included.

Introduction

The determination of 3-D motion of a rigid body from range data is important for artificial intelligence applications in robotic vision. Motion estimation can be useful in scene analysis, in segmentation of a scene containing several moving objects, or, for motion prediction where collision avoidance is desired. The estimation of three dimensional motion requires two stages of processing: assuming that range data is available at two separate time instances, corresponding points on a rigid body at each time instant are first determined. This is a rather difficult problem to solve in practice, but matching stereo pairs, for example, has been an ongoing area of research for some time. This correspondence problem will not be presented here, but rather the second processing step : the determination of the rotation and translation given these 3-D correspondences.

On a first glance, solving for the motion parameters seems straight-forward. However, it will be shown that numerical accuracy as well as computation time are tied closely to the method of solution used. This paper explores both linear and non-linear approaches to solution and examines alternative ways in which to decompose a general 3-D motion (containing both rotation and translation). Included in this discussion is an interesting parameterization of general 3-D motion known as the screw decomposition. The subsequent section discusses conditions in which the different methods fail. Ways of improving some of the solution methods are then given followed by a presentation of simulation results comparing the methods in a quantized three dimensional environment, one that may be encountered in data acquisition methods employing sparse range maps. The 3-D methods not only offer improved numerical accuracy and computational advantages over 2-D methods[1], but also require fewer point correspondences (a minimum of three) allowing less sophisticated matching algorithms[2]. Furthermore, these methods can also be applied to image data where stereo camera pairs are used.

In the stereo case, a corresponding pair of points in the image plane are related to a 3-D point in space through a perspective transformation[2]. With stereo camera alignment such that only horizontal displacement exists between them (the epipolar constraint), matching may be accomplished in a relatively simple and efficient manner[3]. Here, the perspective transformation reduces to a simple triangulation[4].

Basic Mathematical Formulation

In order to describe the problem in mathematical terms, we define a general displacement by the equation

$$p' = Rp + t \qquad (1)$$

where p and p' are the positions of the rigid body at time instances T_1 and T_2, respectively, R is a 3-dimensional rotation matrix (3x3) and t is the translation vector. In component form, then,

$$\begin{bmatrix} p'_x \\ p'_y \\ p'_z \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

Direct Linear Method

From the above, one may initially observe that there are twelve parameters to be solved for, namely R (9 parameters) and t (3 parameters). Accordingly one can express the above as three 4-dimensional systems of linear equations :

$$\begin{bmatrix} p_{1x} & p_{1y} & p_{1z} & 1 \\ p_{2x} & p_{2y} & p_{2z} & 1 \\ p_{3x} & p_{3y} & p_{3z} & 1 \\ p_{4x} & p_{4y} & p_{4z} & 1 \end{bmatrix} \begin{bmatrix} r_{11} \\ r_{12} \\ r_{13} \\ t_x \end{bmatrix} = \begin{bmatrix} p'_{1x} \\ p'_{2x} \\ p'_{3x} \\ p'_{4x} \end{bmatrix} \qquad (1a)$$

$$\begin{bmatrix} p_{1x} & p_{1y} & p_{1z} & 1 \\ p_{2x} & p_{2y} & p_{2z} & 1 \\ p_{3x} & p_{3y} & p_{3z} & 1 \\ p_{4x} & p_{4y} & p_{4z} & 1 \end{bmatrix} \begin{bmatrix} r_{21} \\ r_{22} \\ r_{23} \\ t_y \end{bmatrix} = \begin{bmatrix} p'_{1y} \\ p'_{2y} \\ p'_{3y} \\ p'_{4y} \end{bmatrix}$$

$$\begin{bmatrix} p_{1x} & p_{1y} & p_{1z} & 1 \\ p_{2x} & p_{2y} & p_{2z} & 1 \\ p_{3x} & p_{3y} & p_{3z} & 1 \\ p_{4x} & p_{4y} & p_{4z} & 1 \end{bmatrix} \begin{bmatrix} r_{31} \\ r_{32} \\ r_{33} \\ t_z \end{bmatrix} = \begin{bmatrix} p'_{1z} \\ p'_{2z} \\ p'_{3z} \\ p'_{4z} \end{bmatrix}$$

which implies that four correspondences are needed to solve the above: (p_1, p'_1), (p_2, p'_2), (p_3, p'_3), and (p_4, p'_4) where $\{(p_i, p'_i) : p_{ix}, p_{iy}, p_{iz} \rightarrow p'_{ix}, p'_{iy}, p'_{iz}\}$. The condition of the 4-dimensional square matrix in (1) is dependent entirely on the positions of the points chosen on the rigid body. Here, four correspondences are needed. For the matrix in (1) to have full rank, then, the four points must not be coplanar. Note that the numerical stability is independent of the choice of motion, a desirable property.

Method Based on Translation Invariants

We can associate a unit vector, $\hat{m}$, with any 3-D line which we define as

$$\hat{m} = \frac{p_2 - p_1}{\| p_2 - p_1 \|} \qquad (2)$$

to be the relative orientation of points p_1 and p_2 on a rigid body. Now if the body undergoes a pure translation, these "m" parameters do not change. Only when the body rotates are the "m" vectors transformed. Taking two points on the body, p_i and p_{i+1} (which undergo the same displacement due to rigidity of the body) to move to respective positions p'_i, p'_{i+1}:

$$p'_i = Rp_i + t \qquad (3)$$

$$p'_{i+1} = Rp_{i+1} + t$$

Subtraction eliminates the translation, t, and division by a constant yields

$$\frac{p'_{i+1} - p'_i}{\| p'_{i+1} - p'_i \|} = R \frac{p_{i+1} - p_i}{\| p_{i+1} - p_i \|}$$

which from (2) is

$$\hat{m}'_i = R\,\hat{m}_i \qquad (4)$$

for all point correspondences $1 \leqslant i \leqslant 3$.

We can solve three 3x3 systems directly for R and obtain t by substitution into (1). Note that the solution of the systems of linear equations are now of lower dimension (three) than in equation 1a. Finally, in order for a unique solution to exist, the 3x3 matrix of "$\hat{m}$" vectors must be of full rank. This means that the three "$\hat{m}$" vectors must not be coplanar. As a result, four point correspondences are needed (same restrictions as method 1) and that any motion (combination of R and t) are acceptable.

It must be pointed out that a general 3-D displacement can be described in far fewer than 12 parameters. Indeed the rotation matrix R is an orthogonal matrix which can be expressed in terms of a rotation axis $\hat{n} = [\, n_x, n_y, n_z \,]^t$ and a rotation angle, ϕ, about $\hat{n}$:

$$R = \begin{bmatrix} (n_x^2 - 1)(1 - \cos\phi) + 1 & n_x n_y(1 - \cos\phi) - n_z\sin\phi & n_x n_z(1 - \cos\phi) + n_y\sin\phi \\ n_y n_x(1 - \cos\phi) + n_z\sin\phi & (n_y^2 - 1)(1 - \cos\phi) + 1 & n_y n_z(1 - \cos\phi) - n_x\sin\phi \\ n_z n_x(1 - \cos\phi) - n_y\sin\phi & n_z n_y(1 - \cos\phi) + n_x\sin\phi & (n_z^2 - 1)(1 - \cos\phi) + 1 \end{bmatrix} \qquad (5)$$

Since $\hat{n}$ is a unit vector, R can be expressed in terms of 3 parameters which, in addition to the translation vector yields six parameters that describe a general displacement in 3-D space. Alternatively, then, the R matrix and t vector can be determined by solving for these 6 parameters rather than 12 as in the previous case. The price paid is that the displacement is not linear in $\hat{n}$, ϕ, and t and solution is less straightforward.

Although from (5) it may initially seem quite forbidding to solve for the rotation parameters to get R, two successful nonlinear approaches have made it possible to express the rotation axis $\hat{n}$ and the rotation angle ϕ directly in closed form in terms of the point correspondences. this also yields great computational advantages over solving system (1a) since no equations need be solved.

Adapted Spherical Projection Method

Now eq. (4) is in the same form as the method used by Yen and Huang in[5] for the pure rotation case since these $\hat{m}$ vectors undergo identical transformations as projection vectors on the unit sphere. The rotation axis, $\hat{n}$, can then be obtained directly from two noncollinear "$\hat{m}$" unit vectors, $\hat{m}_1$ and $\hat{m}_2$.

$$\text{if} \quad \hat{m}'_1 = R\,\hat{m}_1$$

$$\hat{m}'_2 = R\,\hat{m}_2$$

$$\text{then } \hat{n} = (\hat{m}_1' - \hat{m}_1) \times (\hat{m}_2' - \hat{m}_2). \qquad (6)$$

Let $\hat{m}$ and $\hat{m}'$ represent one of the line correspondence pairs. The rotation angle ϕ can then be determined (knowing $\hat{n}$) by first solving the scalar equation

$$(\hat{n} \cdot \hat{m})^2 = (\hat{n} \cdot \hat{m}')^2 = \left| \frac{\hat{m} \cdot \hat{m}' - \cos\phi}{1 - \cos\phi} \right| \qquad (7)$$

If $k_1 \equiv (\hat{n} \cdot \hat{m})^2$ and $k_2 \equiv \hat{m} \cdot \hat{m}'$ a rearrangement of the above gives

$$\cos\phi = \frac{(k_2 - k_1)}{(1 - k_1)} \qquad (8)$$

Finally, $\sin\phi$ is found by

$$\sin\phi = k(1 - \cos^2\phi)^{1/2} \qquad (9)$$

where $k = \pm 1$ according to the sign of $[\hat{n} \cdot (\hat{m} \times \hat{m}')]$[5]. From $\cos\phi$, $\sin\phi$, and $\hat{n}$, the rotation matrix, R (eq. 5) can be computed directly. Finally, the translation vector t can be solved for by substitution since R is known, i.e. $t = p' - Rp$. The entire procedure requires only additions and multiplications and obviously much fewer manipulations than in the solutions of (1) and (4).

It is no longer true here that the numerical stability is purely a function of the point set chosen. Of course, the points must not be collinear in order to uniquely specify the position of the rigid body implying that three point correspondences are needed. However, by observing (7), if $\cos\phi = 1$ (pure translation case) the quantity k_1 becomes unbounded. Thus, as the motion degenerates to a pure translation, this method's accuracy decreases significantly.

The Screw Decomposition

As a somewhat different approach, a general 3-D displacement can be decomposed into a different set of parameters than the six used previously: n_x, n_y, ϕ, t_x, t_y, and t_z (since $\hat{n}$ has unit magnitude n_z is determined by n_x and n_y). These are six "screw" parameters s_x, s_y, s_{0x}, s_{0y}, θ, and d .

It is worthwhile to comment that that in the previous case the rotation, ϕ, was about an axis (with direction $\hat{n}$) through the origin of the coordinate system. Such a representation, however, is often unnatural in the real world since many rigid bodies (aircraft, missiles, footballs, etc.) appear to rotate about an axis *not* through the origin of the frame of reference, but rather seem to undergo a "screw motion" about an axis parallel to the direction of translation. In other words, the rotation axis need not be fixed to pass through the origin but can instead be displaced a distance from the origin, even to pass through the object itself. Moreover, the translation can be defined as occurring purely in the direction of this screw axis. The fact that a general 3-D displacement can be represented as a *unique* screw displacement is a statement of Chasles' Theorem[6]. It turns out that this screw decomposition is characterized by six parameters,

$$s_x,\ s_y,\ s_{0x},\ s_{0y},\ \theta \text{ , and } d \tag{10}$$

where $[s_x, s_y, s_z]^t \equiv \hat{s}$ is a unit vector representing the orientation of the screw axis, $[s_{0x}, s_{0y}, s_{0z}]^t \equiv s_0$ is the perpendicular distance of the screw axis to the origin, θ is the screw rotation angle about $\hat{s}$ located at s_0, and d is the translation along $\hat{s}$. Note that s_0 and $\hat{s}$ are perpendicular and $\hat{s}$ is a unit vector so $s_0 \cdot \hat{s} = 0$ and $\hat{s} \cdot \hat{s} = 1$ are true so only four parameters are needed to describe $\hat{s}$ and s_0. The inclusion of d and ϕ brings the total number of " screw parameters" to six.

First, the relationship between a screw displacement and the general displacement given in (1) must be determined. This will result in a solution method that solves first for these screw parameters and then expresses R and t defined earlier in terms of quantities in (10). Before presenting this solution method, the relationship between the two displacements will be shown. Recall that a screw displacement is a rotation of θ about an axis, $\hat{s}$, located s_0 from the origin (possibly through the body itself) followed by a translation, d, along this axis. The displacement p to p' can be expressed in a similar manner as previously if we define

$$q = p - s_0 \tag{11}$$
$$q' = p' - s_0$$

resulting in a displacement q to q' about an axis through this new origin at - $[s_0]$ followed by the screw translation. We have

$$q' = R_s\, q + d\hat{s}$$

or

$$(p' - s_0) = R_s\,(p - s_0) + d\hat{s}$$

so

$$p' = R_s p + (I - R_s)\, s_0 + d\hat{s} \tag{12}$$

From (1), $p' = Rp + t$ so the relationships are

$$R = R_s \text{ , } t = (I - R_s)\, s_0 + d\hat{s} \tag{13}$$

From (13) note that the screw rotation matrix R_s is identical to R so the parameters ϕ and $\hat{n}$ correspond to θ and $\hat{s}$ in the screw case. Also, the translation vector, t, is composed of two components, one corresponding to the translation along the screw axis and another corresponding to the change in location of the screw axis relative to the origin of the original system.

A solution method for finding $\hat{s}$, s_0 , and θ has been proposed in Bottema and Roth[6] based on Rodrigues' Formula, a not widely known kinematical result: given point correspondences (p, p'), (q, q'), and (r, r') first the screw axis, $\hat{s}$, is found directly from

$$\tan\left(\frac{\theta}{2}\right)\hat{s} = \frac{[(r' - q') - (r - q)] \times [(p' - q') - (p - q)]}{[(r' - q') - (r - q)] \cdot [(p' - q') + (p - q)]} \tag{14}$$

and normalizing the left-hand side. Also, $\cos\theta$ and $\sin\theta$ can be determined from $\tan\left(\frac{\theta}{2}\right)$, the magnitude of the right-hand side, by the trig identities

$$\cos\theta = \frac{1 - \tan^2\frac{\theta}{2}}{1 + \tan^2\frac{\theta}{2}}$$

$$\sin\theta = \frac{2\tan\frac{\theta}{2}}{1 + \tan^2\frac{\theta}{2}} .$$

Then from [6] and using one of the point correspondences,

$$s_0 = \tfrac{1}{2}\left| p + p' + \frac{(\hat{s} \times (p' - p))}{\tan\frac{\theta}{2}} - \hat{s} \cdot (p + p')\,\hat{s} \right| \tag{15}$$

Finally, $d = \hat{s} \cdot (p' - p)$ since d is the displacement projected along the screw axis. R and t can be found by substituting these quantities into (13). Note that in (14) $\hat{s}$ becomes unspecified as θ approaches zero (pure translation case) and becomes unbounded as θ approaches 180 degrees. Also, the location, s_0, of $\hat{s}$ in (15) becomes unbounded as θ approaches zero.

Up to this point several methods have been presented that solve for rotation and translation parameters. Two of these are linear methods, the direct method (1), (4x4 systems) and the translation invariant method (4) which first involved solving 3x3 systems of "m" parameters for R and then substituting for t. The other two methods were nonlinear and did not involve solving systems of equations, that of "adapted spherical projection" and that of "screw decomposition". Before comparing the numerical differences between these methods, it is first pertinent to state the conditions under which each of the methods fail. There are two criteria which affect the numerical properties of the solutions. The first consideration is the choice of point correspondences obtained from the matching process. The second is the behavior of each method under different possible motions. Obviously, a unique solution for any of the methods require that the set of points chosen be not collinear. Trivially, three points lying along the same line do not specify the object's position in space so an infinite number of solutions exists.

Improving the Linear Methods

A key concept to realize from the preceding discussion is that the linear methods, although somewhat more expensive computationally, (which may be critical for real time implementation) have numerical accuracy independent of the motion (R,t). The drawback seemed to be insuring the proper selection of point correspondences. Also, the linear methods use four rather than three points (the required minimum for uniqueness) . To overcome this unnecessary extra information requirement (the fourth point) and at the same time obtain well conditioned systems (and more accurate results) from three points, a "pseudo-correspondence" can be constructed artificially. This point is, of course, uniquely determined by the other three points on the body and is chosen such that the solution matrix be well-conditioned. In the case of points ("direct linear method") the fourth point is found by taking the cross product of two lines formed by the other three points and then recording a position of a point on that line. In the case of "m" vectors ("translation invariant method") the same procedure can be adopted by finding a third $\hat{m}$ orthogonal to the other two. One would expect better results in the latter case since (1) "m" vectors are normalized and (2) the system is of lower dimension (see simulation results). These variations above will be respectively called "improved direct linear method" and "improved translation invariant method". Note that the point correspondences themselves need not be used in the solution of the motion, but rather a unique basis description of the point configuration with desirable properties (e.x. orthonormal) can be used to actually perform the computation.

Overdetermined Systems

It may be possible in practice to obtain a greater number of matches (point correspondences) than the minimum necessary (three in all cases). Assuming that a large number of correspondences were obtained with equal accuracy one can ahead of time determine the subset of three points that will most probably yield the best accuracy and have superior performance over some technique that averages out the many possible solutions in a seemingly arbitrary way. This is simply the subset of points forming a matrix (for the linear methods) having lowest condition number (i.e. least "singular") over all subsets of three points. If a linear equation solver is used in which the decomposition stage computes a good estimate of the condition number of the matrix such as DECOMP in [7], then a number of candidate point sets can first be decomposed and the point set with the lowest condition number be used. This has been simulated in the next section along with other methods described previously.

Simulation Results

The methods described in earlier sections have been simulated and the results for a quantized three dimensional space of 1024 by 1024 by 1024 are shown here. Corresponding results at 512 (cubed) and 256 (cubed) quantizations yield respectively about double and quadruple the average errors obtained for the "1024" case. For the simulation a general motion consisting of a rotation of 23 degrees about an axis with direction (0.7, 0.5, 0.51) and a translation of (63, 35, -150) in this quantized environment was used. Many other motions were also tried, all with similar results to this example (cases that approached pure translation violate this assertion for nonlinear methods as explained earlier on) . Average errors over 1000 sets of randomly generated points and their correspondences are computed (over the 9 "matrix" rotation components and 3 translation components) . The reasoning behind this use of averaging is to measure the performance of a solution method independent of the sets of point correspondences chosen. Worst case results (over the 1000 "trials") are presented and tend to correspond to the case described earlier involving "nearly collinear" points.

The results are shown in the following two tables. The first table consists of a comparison between three variations of the "direct linear method" :

(1) the straight-forward approach first mentioned (that uses *four* correspondences),

(2) same as above except four of the correspondences chosen were from an "overdetermined" case where *five* correspondences were available, and

(3) the "improved direct linear method" described (that uses only *three* point correspondences and generates a "pseudo" correspondence.

The second table lists the improved versions of the linear methods as well as the two nonlinear methods discussed. Note that the "improved translation invariant method" gives the best results with even the worst case being somewhat reasonable. Each of the following methods use equal amounts of a priory information (three point correspondences).

Table I

Method Used	Avg % errors in Rotation matrix components	Avg % errors in translation components	" " of worst component
direct linear method	3.10	22.42	273.0
same but "overdetermined"	0.464	1.79	35.6
"improved" version	0.337	1.45	21.7

Table II

Method Used	Avg % errors in Rotation matrix components	Avg % errors in translation components	" " of worst component
improved direct linear	0.337	1.45	21.7
improved translation invariant	0.279	1.11	12.7
adapted spherical projection	4.27	9.33	64.9
screw decomposition	2.65	13.7	173

Conclusion

From the results obtained these methods performed very well (and at very low computational cost) yielding average errors (for the best method) of less than one percent for the motion shown. This is encouraging since this best method ("improved translation invariant method") has accuracy independent of the motion to be solved for. However, the true "bottleneck" lies in the problem in obtaining these point correspondences in the first place which has not been discussed here. Existing procedures for finding correspondences (the matching problem) are slow and not completely reliable. In the future, simulation results from 3-D points derived from triangulation of stereo image pairs will be investigated to determine whether these 3-D methods can be applied to two-dimensionally acquired data.

Acknowledgement

This work was supported by National Science Foundation Grant ECS-81-12082.

References

[1] R.Y. Tsai, T.S. Huang, "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces". Report R-921, Coordinated Sciences Laboratory, University of Illinois.

[2] Kak, A. *Depth Perception for Robots* , Technical Report, Purdue University.

[3] S.Barnard, M.A. Fischler, *Computational Stereo* , Computing Surveys, Vol. 14, No. 4, December, 1982.

[4] Ballard and Brown, *Computer Vision* , Prentice Hall.

[5] B.L. Yen and T.S. Huang, "Determining 3-D Motion and Structure of a Rigid Body Using the Spherical Projection". Computer Vision, Graphics and Image Proc. Vol. 21, 1983.

[6] O.Bottema, B.Roth, *Theoretical Kinematics* , North Holland Pub. Co. N.Y. , 1979.

[7] Forsythe, Malcolm, Moler, *Computer Methods for Mathematical Computations* , Prentice Hall.