Optimal Visual Motion Estimation: A Note

Minas E. Spetsakis and Yiannis Aloimonos

Abstract— **We analyze the problem of estimating 3-D motion in an optimal manner using correspondences of features in two views. The importance of having an optimal estimator is twofold: first, for the estimation itself and, second, for the bound it offers on how much sensitivity one can expect from a two-frame, point-based motion algorithm. The optimal estimator turns out to be nonlinear, and for that reason, we developed techniques that provide very good initial guesses for the iterative computation of the optimal estimator.**

Index Terms— **Correspondence, structure from motion, 3-D motion estimation.**

I. Introduction

In this correspondence, we develop and analyze an optimal estimator for the structure-from-motion problem[1] under the assumption of Gaussian noise. The issue of optimal estimation is becoming quite important lately because of the potential of applications in robotics. It is clear that we need to compute solutions to robot vision problems as efficiently and robustly as possible (in other words, the "best" solution, in some sense). However, we also need to know how good the best is. If the best is unstable, then we should look for a new direction in research. If the best is stable, it is not news until an efficient way to compute it is developed.

The formalism of the problem, as found in most of the literature, is geared toward studies of uniqueness. In these studies, an ideal situation is assumed, and an algorithm is designed. This approach gave a nice theoretical framework on which to build. However, the noise-sensitivity problem was far from solved. Therefore, the next step is to acknowledge the existence of noise, decide on a model for it, and then formulate the problem as a statistical estimation. The result then will be an estimate that is optimal under the assumption of the noise model. There are several ways to obtain optimal estimates in the sense of being unbiased, possessing minimum variance, being asymptotically normal, or any combination of these. The most popular is the maximum likelihood estimator. Among the desirable features of the maximum likelihood estimator is its convergence properties, where for large samples, the estimated quantity is normally distributed, and among other asymptotically normally distributed estimators, this one has the least asymptotic variance. This estimator can also very often be proved unbiased. Since the Gaussian assumption for the noise is almost always present, the maximum likelihood binds very well with the least squares method.

We picked the maximum likelihood estimator to build our optimal estimator as the most promising among the statistical inference techniques. Of course, any other estimator that could give better results could replace this one, but most probably, the better one is going to be an estimator tailored to the needs of the specific problem at hand. After the first version of this correspondence was published [15], papers with similar results appeared in the literature [2], [12].

Manuscript received November 3, 1989; revised March 31, 1992. Recommended for acceptance by Associate Editor N. Ahuja.

M. E. Spetsakis is with the Department of Computer Science, York University, North York, Canada M3J 1P3.

Y. Aloimonos is with the Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, MD 20742.

IEEE Log Number 9201898.

[1]The structure-from-motion problem has received a lot of attention in the past several years [21]. Depending on context, this problem is also known as passive navigation, the kinetic depth effect, or relative orientation, with a slightly different meaning each time in robot navigation, psychology, or photogrammetry, respectively. Here, we deal with the problem mainly in the context of computer vision; in other words, we are looking for algorithms that can be implemented on machines, can work without manual intervention, and are general enough for a wide spectrum of applications.

The problem of structure from motion has been studied for both the differential (small motion [1], [8], [4], [21], [11]) and the discrete (large motion [18], [19], [22], [20], [7], [10]) case. Here, the formalism is the one of the discrete case, but the results can be applied to the differential case as well. The problem in geometric terms can be expressed as follows: Given the projections of a number of points on the two image planes and the knowledge of the correspondence (i.e., which points in the two frames are the projections of the same 3-D point), recover the parameters of the rigid motion between the two coordinate systems.

With respect to the uniqueness issues associated with the problem, recent research [5] investigated the minimum number of points required for the problem to be solvable as well as the number of distinct solutions. Research on uniqueness concentrates on trying to obtain a closed-form solution for the five motion parameters (direction of translation and rotation). Such a closed-form solution was developed by Longuet-Higgins [10] and further analyzed by Tsai and Huang [19], [18]. Because this solution, which is based on eight points and uses linear least squares for more points, is not robust in the presence of noise, researchers have recently concentrated on the development of algorithms that exhibit robustness properties [22], [7], using many points and more elaborate optimization techniques. In [22], the maximum likelihood principle is applied on all the parameters of the problem, and in [7], an expression from [18] is minimized using an elaborate technique.

II. Overview of the Approach

The main advantage of using Gaussian assumption for the noise is that the maximum likelihood estimation becomes a least squares minimization. Unfortunately, the weights suggested by this technique are not constant; they depend on the motion parameters. As will become evident later, this makes the minimization more expensive because the program has to go through all the points in the image at each iteration. To speed up the process, we devised a technique that finds a suboptimal solution very quickly that, used as an initial guess for the optimal estimator, leads to a quick convergence. Therefore, the result is an efficient algorithm for the optimal estimation of motion.

The suboptimal solution can be found by setting the weights in the optimal estimator to 1. Then, we can factor out the motion parameters and do the minimization without having to go through all the points in each iteration; the information from the image points is coded in a 9×9 matrix, which is then operated on to find the suboptimal solution. A somewhat similar approach for a suboptimal solution was used by Jerian and Jain [9] and Horn [7].

III. Previous Work and Statement of the Problem

Several algorithms dealing with motion estimation from discrete frames have been published. The most notable of them was presented and analyzed in [18] and [10]. We summarize it here and then proceed to the noise analysis.

The imaging geometry is the usual one: coordinate system $OXYZ$, image plane $Z = 1$, nodal point O. A point P in 3-D is represented by its position vector $[X \;\; Y \;\; Z]^T$ and its image on the image plane by $P \cdot \frac{1}{Z} = \left[\frac{X}{Z} \;\; \frac{Y}{Z} \;\; 1\right]^T$ in 3-D coordinates or $[x \;\; y]^T$ in image plane coordinates.

The vector $p = \frac{P}{\|P\|}$ contains as much information as $P \cdot \frac{1}{Z}$ and has the advantage of constant length. This will be useful later in doing least squares; otherwise, the points far away from the center of the image get unfairly high weight, which is, in general, different from what a sophisticated camera model would suggest. Furthermore, when an object point P rotates to $R \cdot P$, then the corresponding image

points are p and $R\cdot p$ if they are normalized to unity and p and $\frac{R\cdot p}{\hat{z}^T\cdot R\cdot p}$ if they are not ($\hat{z}$ is the unit vector along the Z axis). This simplifies things a lot. Note, however, that we do not use spherical coordinates but just a notation that is convenient for mathematical manipulations. Therefore, from now on, we use the unit projection vector p instead of the projection coordinates.

A point P in 3-D that projects on p translates by T and then rotates by R (R is a rotation matrix) to P', which in turn projects to p'. The following relation then holds:

$$P' = R(P+T) \Rightarrow R^T\cdot P' = P+T \Rightarrow$$
$$T\times(R^T\cdot P') = T\times P \Rightarrow P\cdot\left(T\times(R^T\cdot P')\right) = 0$$

or $[P, T, R^T\cdot P'] = 0$ where $[\cdot,\cdot,\cdot]$ is the triple scalar product. Dividing by $\|P\|\cdot\|P'\|$, we get $[p, T, R^T\cdot p'] = 0$ or

$$p^T\cdot E\cdot p' = 0 \tag{1}$$

where $E = T_S\cdot R^T$

$$T_S = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$$

Equation (1) is a linear equation with unknowns: the elements of E. If we take at least eight such equations, we can almost always recover the motion parameters [18]. To increase the stability, we can take more than eight points and do least squares to minimize a quadratic of the form

$$x^T\cdot A\cdot x \rightarrow \min \tag{2}$$

where x is a 9-D vector, where each element is an element of E (its columns, one on top of the other, form x), and A is a matrix that depends on the various pairs of points p_i, p'_i.

Least squares is the easiest method we can use, but it requires the variables (the elements of the vector x) to be independent. Here, this is not the case; the solution that least squares finds, without taking into consideration the dependency, does not represent a matrix E that is decomposable into T_S and R^T. Even if, from the solution that minimizes (2), we find a matrix E that is nearest to being decomposable, this might be far from minimizing (2) in the sense of finding R, T that do so.

Another problem is the physical interpretation of what we minimize. Unless x is decomposable to R, T, then there is no physical interpretation of the quantity we minimize.

Therefore, two things need to be done: First, use constrained minimization for (2), and second, find what the quantity we minimize stands for. Finding the physical interpretation of the quantity we minimize will help develop the optimal estimator, and the constrained minimization will provide a good initial guess for it. Let us now introduce the error in correspondence in our calculations.

A point P_1 moves to P_2 with rigid motion $P_2 = R(P_1+T)$. The correspondence algorithm matches it incorrectly with $P'_2 = P_2 + n$ or $P'_2 = R(P_1+T)+n$, where n is the error vector. Proceeding as before, we finally get

$$p_1^T\cdot E\cdot p'_2 = \left[p_1, T, R^T\frac{n}{|P_2|}\right] \text{ or}$$
$$p_1^T\cdot E\cdot p'_2 = [p_1, T, n'] \tag{3}$$

where

$$n' = R^T\frac{n}{|P_2|}. \tag{4}$$

The left-hand side of (3) is what we minimize in (2). Therefore, this minimization process minimizes a function of the correspondence error. The right-hand side of (3) equals

$$(p_1\times T)\cdot n'. \tag{5}$$

First notice that we minimize the component of the error that is parallel to $p_1\times T$. The other component is irrelevant to the estimation of motion and affects only the estimation of the structure of each point; hence, depth estimation for each of these points is at the mercy of the error in the pair of its projections. Needless to say, trying to minimize both components of the error is impossible. This difficulty can be solved with a many-frame formulation of the problem [17]. Second, far-away points have less weight because, in (4), $\|P_2\|$ is in the denominator. What we actually minimize is one component of the image of the noise vector. Both of the above are natural, and both of them are to be expected.

One of the difficulties inherent in estimating the motion is related to the size of the object observed. When the object is both small and almost planar, then pure translation and pure rotation may create very similar flow patterns or correspondence pairs [1]. This phenomenon appears here as well but in a slightly worse form. In (5), the error is multiplied by the sine of the angle between p_1 and T. When the field of view is small, then the vectors p_i of the points form a tight bundle. Then, a choice of T somewhere between them makes both the sine of the angle between T and the points very small. Since this sine is multiplied by the error, the result is a small number. If the noise is sufficiently large, then the solution of T is biased toward being pointed to the object. Part of the blame here goes to the sine that appears in (5).

As a conclusion, we can say the following:

- No matter how many points we use, we cannot reduce the error in structure estimation using *two frames*. This problem cannot be cured with two frames.
- When the field of view is small and there is noise, the translation is biased toward the observed object. Ultimately, this means high error in the output because this estimator is biased.

We now discuss the minimization, both unconstrained and constrained, and then we discuss the optimal estimation.

IV. A Useful Result

We present a result related to the well-known algorithm by Tsai and Huang [18] and Longuet-Higgins [10] that is going to be useful in the next section.

Definition: We define the *vector* of a 3×3 matrix E to be $\mathbf{V}(E)$, which is a vector of dimension 9 whose elements are the same as the elements of the matrix E and ordered so that they are the columns of E one on top of the other.

Tsai and Huang [18] developed an algorithm that finds T and R, given a matrix E for which there exists a vector T and a matrix R such that

$$E = T_S\cdot R^T$$

where

$$T_S = \begin{bmatrix} 0 & t_3 & -t_2 \\ -t_3 & 0 & t_1 \\ t_2 & -t_1 & 0 \end{bmatrix}, \quad T = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}.$$

They proved that there are two solutions, and the algorithm can find both. Furthermore, the algorithm is very stable in the presence of noise, partly making up for the extreme instability of the process of finding the matrix E. Overall, though, the algorithm behaved poorly due to the difficulties in finding E [18], and there is no solution when the points in the world lie on some critical surfaces [18], [6].[2] Below,

[2] In dealing with noise, this is important only when almost all 3-D points are on or close to a critical surface.

$$x = \begin{bmatrix} t_3r_2 - t_2r_3 \\ t_1r_3 - t_3r_1 \\ t_2r_1 - t_1r_2 \\ t_3r_5 - t_2r_6 \\ t_1r_8 - t_3r_7 \\ t_2r_7 - t_1r_5 \\ t_3r_8 - t_2r_9 \\ t_1r_9 - t_3r_7 \\ t_2r_7 - t_1r_8 \end{bmatrix} = \begin{bmatrix} r_1 & 0 & 0 & r_2 & 0 & 0 & r_3 & 0 & 0 \\ 0 & r_1 & 0 & 0 & r_2 & 0 & 0 & r_3 & 0 \\ 0 & 0 & r_1 & 0 & 0 & r_2 & 0 & 0 & r_3 \\ r_4 & 0 & 0 & r_5 & 0 & 0 & r_6 & 0 & 0 \\ 0 & r_4 & 0 & 0 & r_5 & 0 & 0 & r_6 & 0 \\ 0 & 0 & r_4 & 0 & 0 & r_5 & 0 & 0 & r_6 \\ r_7 & 0 & 0 & r_8 & 0 & 0 & r_9 & 0 & 0 \\ 0 & r_7 & 0 & 0 & r_8 & 0 & 0 & r_9 & 0 \\ 0 & 0 & r_7 & 0 & 0 & r_8 & 0 & 0 & r_9 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -t_3 \\ t_2 \\ t_3 \\ 0 \\ -t_1 \\ -t_2 \\ t_1 \\ 0 \end{bmatrix}$$

we rephrase the algorithm in our notation, and we prove that the R, T that their algorithm finds are such that

$$\|\mathbf{V}(E) - \mathbf{V}(M)\| = \min$$

where $M = T_S \cdot R^T$.

Algorithm: Let E be a 3×3 matrix. We find T, R as follows: If the singular value decomposition (SVD) of E is

$$E = U \begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \sigma_3 \end{bmatrix} V^T \quad . \quad 0 \le \sigma_1 \le \sigma_2 \le \sigma_3$$

then T is parallel to the first column U_1 of U and $\|T\| = \frac{\sigma_2+\sigma_3}{2}$. Matrix T_S has two degenerate singular vectors and one parallel to U_1.[3] Therefore, one of its possible SVD's is

$$T_S = U \begin{bmatrix} 0 & & \\ & \|T\| & \\ & & \|T\| \end{bmatrix} V_t^T.$$

Then, R is

$$R^T = V^T \begin{bmatrix} s & & \\ & s_1 & \\ & & s_1 \end{bmatrix} V_t$$

where $s = \det(V_t) \cdot \det(V)$ and $s_1 = \pm 1$.

Theorem: The R, T computed above satisfy $\|\mathbf{V}(E) - \mathbf{V}(M)\| = \min$, where $M = T_S \cdot R^T$.

Proof: See [15].

V. Solving the Constraint Minimization Problem

The mathematical problem at hand is to find a 9-D vector x such that

$$x^T \cdot A \cdot x \to \min$$

and the matrix, whose vector is x, to be decomposable to R, T, as described above. The constraint is clearly nonlinear and very difficult to be written down analytically. We describe here two methods to treat the problem: One is a variation of Newton's method, and the other is a decomposition of the problem into two parts, thus reducing the dimensionality. Both of them are efficient.

[3]When two or more singular values are equal, then the corresponding singular vectors are not uniquely defined and are called degenerate.

A. First Method

We present the method along with a proof of convergence. Let $x = \mathbf{V}(E)$

$$E = T_S \cdot R^T = \begin{bmatrix} 0 & t_3 & -t_2 \\ -t_3 & 0 & t_1 \\ t_2 & -t_1 & 0 \end{bmatrix} \cdot \begin{bmatrix} r_1 & r_4 & r_7 \\ r_2 & r_5 & r_8 \\ r_3 & r_6 & r_9 \end{bmatrix} = \begin{bmatrix} t_3r_2 - t_2r_3 & t_3r_5 - t_2r_6 & t_3r_8 - t_2r_9 \\ t_1r_3 - t_3r_1 & t_1r_8 - t_3r_7 & t_1r_9 - t_3r_7 \\ t_2r_1 - t_1r_2 & t_2r_7 - t_1r_5 & t_2r_7 - t_1r_8 \end{bmatrix}.$$

Therefore x is as defined at the top of this page, or $x = R_b \cdot T_b$, where R_b and T_b are the above matrix and vector, respectively. T_b depends on the three translation parameters. R_b depends on the rotation matrix R, which in turn depends on the three Rodrigues parameters [2] b_1, b_2, b_3 of the rotation. Therefore, x is a function of a 6-D vector $\zeta = [b_1, b_2, b_3, t_1, t_2, t_3]^T$.

The Taylor series expansion of x is

$$x(\zeta + \Delta\zeta) = x(\zeta) + A(\zeta) \cdot \Delta\zeta + 0(\Delta b_1^2) + 0(\Delta b_2^2) + \cdots + 0(\Delta t_3^2).$$

Matrix $A(\zeta)$ is easy to construct. It has, as columns, the derivatives of x with respect to the elements of ζ. The derivatives with respect to t_1, t_2, t_3 are obvious. The derivatives with respect to b_1, b_2, b_3 are $x_i = \frac{1}{2}(R_b + I)B_i(R_b + I) \cdot T_b$, $i = 1, 2, 3$, where the B_i's are

$$B_1 = \begin{bmatrix} 0 & & & & & & & & \\ & 0 & & & & & & & \\ & & 0 & & & & & & \\ & & & 0 & & & +1 & & \\ & & & & 0 & & & +1 & \\ & & & & & 0 & & & +1 \\ & & & -1 & & & 0 & & \\ & & & & -1 & & & 0 & \\ & & & & & -1 & & & 0 \end{bmatrix}.$$

This is similar for B_2, B_3.

The way to achieve convergence is to move in the column space of $A(\zeta)$ so that the quadratic is decreasing in value. This, in general, will lead to values of x that do not satisfy the nonlinear condition. However, if l is the distance we moved in the column space of $A(\zeta)$, then the distance of the nearest vector x that satisfies the nonlinear condition is of order $O(l^2)$. This is why we needed to prove that the Tsai-Huang algorithm finds the nearest vector. If we are not at a local extremum, the quadratic decreases by $O(l)$ and then increases by $O(l^2)$ and, for sufficiently small l, decreases overall. It is easy to see that unless this process goes to a local extremum, it eventually converges to a minimum.

B. Second Method

This is a method that involves gradient descent in a 3-D space. If there is a good guess for the solution, and in this case there is, then we need to move around only locally. We have

$$T_b = \begin{bmatrix} 0 \\ -t_3 \\ t_2 \\ t_3 \\ 0 \\ -t_1 \\ -t_2 \\ t_1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \cdot T = K \cdot T.$$

Then, the quadratic takes the form

$$x^T \cdot A \cdot x = T^T \cdot K^T \cdot R_b^T \cdot A \cdot R_b \cdot K \cdot T = T^T A'^T \qquad (6)$$

where A' is a 3×3 matrix that depends only on the Rodrigues parameters of the rotation.

The value of T that minimizes the quadratic is a vector parallel to the eigenvector of A' that has the smallest eigenvalue. Then, the value of the quadratic is the smallest eigenvalue of A'. (There is a factor of two missing here; when $\|x\| = 1$, then the corresponding $\|T\| = \frac{\sqrt{2}}{2}$. When we minimize $x^T \cdot A \cdot x$, we silently assume $\|x\| = 1$, and when we minimize $T^T A' T$, $\|T\| = 1$. This causes no problem, however.)

Now, the problem is really broken into two: computing the rotation parameters that minimize the smallest eigenvalue of the matrix A' and minimizing a quadratic. The second is just an application of Rayleigh's principle; therefore, there is an easy solution. For the first, it is easy to use a modified Newton's method because we can derive the analytic expression for the derivative. Recall that the minimal value of (6) is the smallest eigenvalue of A'. The derivative of A' with respect to the Rodrigues parameters of R_b is

$$\frac{dA'}{db_i} = \frac{1}{2} \cdot K^T \cdot \Big[R_b^T \cdot A \cdot (R_b + I) \cdot B_i (R_b + I) + (R_b + I)^T \cdot B_i^T (R_b + I)^T \cdot A \cdot R_b \Big] \cdot K.$$

This is an unnecessarily complicated expression and can be simplified as follows: Form the matrix $A_r = R_b^T \cdot A \cdot R_b$, and fix the value of R_b at the current guess. To do gradient descent, we can perturb A_r by pre and postmultiplying by the matrix $R_p(b'_1, b'_2, b'_3)$, which is a function of b'_i's that now serve as unknowns. The initial guess for R_p is the identity matrix (e.g., all the Rodrigues parameters b'_1, b'_2, b'_3 are zero). The expression for the derivative is simplified because we take the derivative at the zero point of the parameters. Therefore, A' is now a function of three new parameters that we can perturb around zero. Thus

$$\frac{dA'}{db'_i} = \frac{1}{2} K^T \cdot \Big[B_i^T \cdot A_r + A_r \cdot B_i \Big] \cdot K.$$

The derivative of the smallest eigenvalue with respect to the ith parameter is

$$\lambda^{(i)} = \phi^T \cdot \frac{dA'}{db'_i} \cdot \phi = 4\, \phi^T \cdot K^T \cdot B_i^T \cdot A_r \cdot K \cdot \phi$$

where ϕ is the eigenvector of the smallest eigenvalue. Using the modified Newton's method, we can find the minimum. This method results in an algorithm that has a quite large basin of attraction, and therefore, it works well if the initial guess is not that good.

VI. Optimality

Here, we are interested in optimal estimation techniques that lead to results that can be studied analytically. The maximum likelihood estimator is best from this point of view. Assuming a Gaussian distribution, it leads to a least-squares formulation on which there is a lot of published work.

The maximum likelihood estimator is formulated as follows: Let $f(p_1, p_2; R, T)$ be the probability density that p_1, p_2 are a correspondence pair when R, T are the motion parameters. Then

$$\prod_i f(p_{i1}, p_{i2}, R, T)$$

is the probability that $\{(p_{i1}, p_{i2}) \mid i = 1 \ldots\}$ are the correspondence pairs. Therefore, if we find R, T that maximize this probability, we have found the most typical solution.

Now, let p_1 be an image unit vector in the first frame and p_2 in the second frame. If p_1, p_2 is a correspondence pair with R, T as motion parameters, then RTp_2 should lie on the plane defined by T and p_1. The error vector on the image has two orthogonal components that are assumed independent identically distributed.

If p_2 is corrupted by an error n', its distance from the plane of T, p_1 is

$$\frac{[(p_1 \times T) \cdot R^T p_2]}{\|p_1 \times T\|} = \frac{[p_1, T, n']}{\|p_1 \times T\|} = \frac{(p_1 \times T) \cdot n'}{\|p_1 \times T\|} = \frac{\epsilon}{\|p_1 \times T\|}.$$

As we see, only the component parallel to the unit vector $\frac{p_1 \times T}{\|p_1 \times T\|}$ affects the distance from the T, p_1 plane. (The direction of this unit vector does not make any difference to the probability distribution because n' is isotropically distributed on the image.) Therefore

$$f(p_1, p_2; R, T) = \alpha e^{-\frac{\left[\frac{\epsilon}{\|T \times p_1\|}\right]^2}{2\sigma^2}}$$

where α and σ are constants and depend only on the noise distribution.

By using the standard procedure for maximum likelihood, we find that we have to minimize the quantity

$$\sum_i \left[\frac{\epsilon_i}{\|T \times p_{1i}\|} \right]^2 \qquad (7)$$

where i is the index for the different points on the image. In the previous paragraphs, we discussed the minimization of

$$\sum_i \epsilon_i^2 \qquad (8)$$

or to be more precise and explicitly incorporate the restriction that $\|T\| = 1$

$$\sum_i \frac{\epsilon_i^2}{\|T\|^2} = \sum_i \frac{\epsilon_i^2}{T^T \cdot T}$$

and now, we see that the optimum has some "weight" factor of $\frac{1}{\|T \times p_{1i}\|^2}$ in the minimization function.

This has two consequences: First, there is some weight in the equations different from 1. This has some small effect on the result. The second consequence is more important. Imagine the following situation: A small object on the z axis translates parallel to the x axis without rotation. Then the translation vector T that minimizes $\sum \epsilon_i^2$ in the presence of noise is parallel to the z axis because most of the p_1's of the object points are very close to the translation vector T. Therefore, (p_1, p_2, T) is very close to zero no matter what the error is.

This way, the solution tends to be parallel to the center of gravity of the p_{i1}'s when the noise level is rather high. This happens because we pick the eigenvector with the smallest eigenvalue, which minimizes (8) but not (7).

To incorporate the factor $\frac{1}{\|T\times p_{1i}\|^2}$ in the computation without increasing the complexity much, we *approximate* the factor with a uniform one on the object and try to minimize the function

$$\frac{T^T A'^T}{\|T \times C\|^2}$$

where C is the center of gravity of the points that constitute the object. This takes the form

$$\frac{T^T \cdot A' \cdot T}{T^T \cdot C' \cdot T} \tag{9}$$

where $C' = I - C \cdot C^T$, $C \cdot C^T$ is the outer product of C with itself, and C is a unit vector. This approximation gives very accurate results in the case of a small object (small viewing angle). There are more details in [16]. In the case of a larger viewing angle, one has to use one of the standard routines that minimize nonlinear functions. The problem with these is that one has to deal with each point in each iteration, which is expensive, compared with the methods discussed above that just construct a 9×9 matrix and iterate on that.

A. Relation to Other Approaches

One proposed approach for motion estimation by Prazdny [12] was based on the following observation: Since we know that the flow pattern of a pure translation is a set of lines converging to a point, we can test different rotation matrices with which to derotate until we find a flow pattern that looks like a pure translation. Prazdny, however, used an overly simplistic measure of similarity to the pure flow pattern. Here, we choose the sum of the squares of the distances of the unit vector T from the planes defined by p_1, p'_2. (p'_2 is p_2 derotated and corrupted by noise.)

In order to find this distance k, we find k such that $T + k\frac{p_1 \times p_2}{\|p_1 \times p'_2\|}$ is coplanar with p_1, p'_2. We have

$$\begin{aligned} 0 &= \left(p_1, p'_2, T + k\frac{p_1 \times p'_2}{\|p_1 \times p'_2\|}\right) \\ &= (p_1, p'_2, T) + \left(p_1, p'_2, k\frac{p_1 \times p'_2}{\|p_1 \times p'_2\|}\right) \\ &= \epsilon + \frac{k}{\|p_1 \times p'_2\|}(p_1, p'_2, p_1 \times p_2) = \epsilon + k \cdot \|p_1 \times p'_2\| \end{aligned}$$

Therefore, $k = \frac{-\epsilon}{\|p_1 \times p'_2\|}$.

Although minimization of k, as defined above, is intuitively a good idea, it is better once again to use a maximum likelihood argument. The variance of k is approximately

$$\sigma_k^2 = \frac{\|T \times p_1\|^2}{\|p_1 \times p'_2\|^2}\tilde{n}$$

where $\tilde{n}$ is proportional to the variance of the error in the image. Proceeding as before, we find that the quantity we want to minimize is

$$\sum \frac{\epsilon_i^2}{\|T \times p_{1i}\|^2}$$

Not surprisingly, it is the same as before.

VII. Experiments

We conducted several comparative experiments, testing both the improvement over the Tsai-Huang algorithm and the convergence of the algorithm to a global minimum with synthetic images. We used a three-stage procedure to converge to a global minimum. First, we used the Tsai-Huang, Longuet-Higgins algorithm to find a guess for the nonlinear suboptimal procedures (both suboptimal procedures performed well; the first had faster convergence, and the second had a wider basin of attraction). This result was fed as a guess for the optimal estimator, which is a standard nonlinear minimization routine. In the diagrams, we plot the Tsai-Huang, Longuet-Higgins algorithm (the curve with the squares), the suboptimal one (which gives the same result as other algorithms using the same norm [7]), and the optimal one (with diamond and circle, respectively). The quantity we plot is error in input versus error in output. (see Figs. 1,2, and 3)

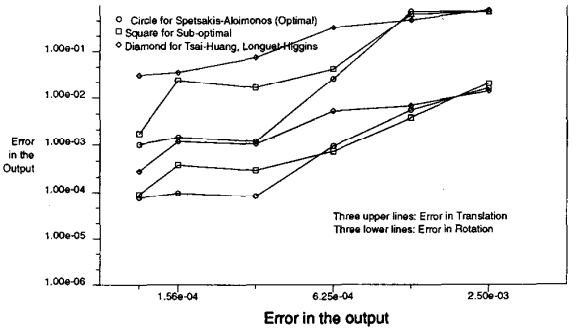

Fig. 1. Response to noisy input for 10 points.

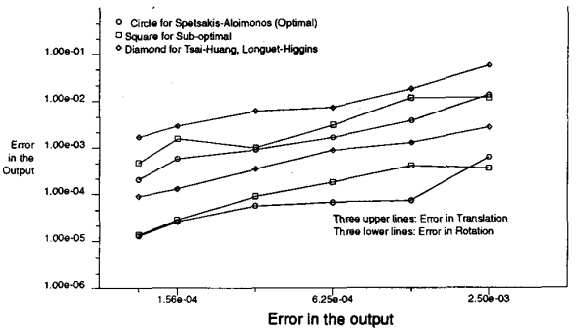

Fig. 2. Response to noisy input for 40 points.

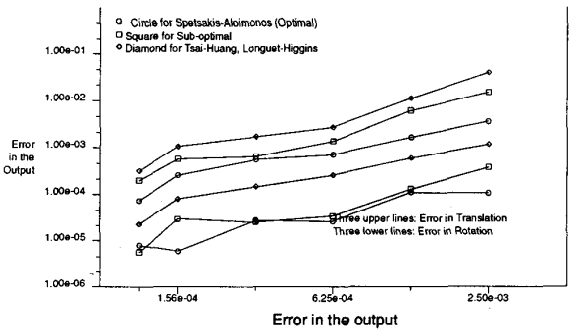

Fig. 3. Response to noisy input for 160 points.

The noise in the output of the algorithm was represented by three numbers: the angle between the two axes of rotation (actual and computed) (phi), the difference in the two rotation angles (theta), and the percent difference of the two translation directions (100 times the sine of their angle). The synthetic object was 30 units away and two units in diameter.

The computation time was less than 1 s for Tsai-Huang and for each iteration of the other algorithms on a Sun 3/280.

VIII. Conclusions

We have presented a method for computing structure from motion in an optimal way. Our contribution lies in showing that our formulation is provably optimal. In addition, we analyzed this nonlinear system in depth so that convergence is fast because we have to deal, for the most part, only once with each point. It has been demonstrated that we can almost always compute this optimal solution efficiently by providing means to compute successively better guesses to the nonlinear procedure that computes the optimal estimate. In addition, our formulation is a framework where past research efforts fit as special cases.

References

[1] G. Adiv, "Inherent ambiguities in recovering 3-D motion and structure from a noisy flow field," in *Proc. IEEE CVPR Conf.*, 1985, pp. 70–77.

[2] J. Aisbett, "An iterated estimation of the motion parameters of a rigid body from noisy displacement vectors," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 12, pp. 1092–1098, 1991.

[3] O. Bottema and B. Roth, *Theoretical Kinematics*. New York: North Holland, 1979.

[4] A. Bruss and B. K. P. Horn, "Passive navigation," *CVGIP* vol. 21, pp. 3–20, 1983.

[5] O. Faugeras and S. Maybank, "Motion from point matches: Multiplicity of solutions," in *Proc. IEEE Workshop Visual Motion: Representation Anal.* (Irvine, CA), 1989, pp. 248–255.

[6] B. K. P. Horn, "Motion fields are hardly ever ambiguous," *Int. J. Comp. Vision*, vol. 1, pp. 259–274, 1987.

[7] ——, "Relative orientation," MIT AI Memo 994, 1988.

[8] B. K. P. Horn and E. Weldon, "Direct motion recovery," in *Proc. IEEE ICCV*, 1987.

[9] C. Jerian and R. Jain, "Polynomial algorithms for structure from motion," in*Proc. 2nd ICCV* (Tarpon Springs, FL), 1988.

[10] H. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," *Nature*, vol. 293, pp. 133–135, 1981.

[11] H. C. Longuet-Higgins and K. Prazdny, "The interpretation of a moving retinal image," in *Proc. Royal Soc. London B*, vol. 208, pp. 385–397, 1984.

[12] J. Philip, "Estimation of three-dimensional motion of rigid objects from noisy observations," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 13, pp. 61–66, 1991.

[13] K. Prazdny, "Determining the instantaneous direction of motion from optical flow generated by a curvilinearly moving observer," *Comput. Vision Graphics Image Processing*, vol. 17, pp. 94–97, 1981.

[14] M. E. Spetsakis, "The geometry and statistics of visual motion," Ph.D. thesis, Comput. Vision Lab., Cent. Automat. Res., Univ. Maryland, College Park, 1989.

[15] M. E. Spetsakis and J. Aloimonos, "Optimal computing of structure from motion using point correspondences in two frames," Tech. Rep. CAR-TR-389, Comput. Vision Lab., Cent. Automat. Res., Univ. Maryland, College Park, 1988.

[16] , "Optimal computing of structure from motion using point correspondences in two frames," in *Proc. Int. Conf. Comput. Vision* 1988, pp. 449–453.

[17] , "A multiframe approach to visual motion perception," *Int. J. Comput. Vision*, vol. 6, pp. 245–255, 1991.

[18] R. Y. Tsai and T. S. Huang, "Uniqueness and estimation of three dimensional motion parameters of rigid objects with curved surfaces," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, pp. 13–27, 1984.

[19] , "Uniqueness and estimation of three dimensional motion parameters of rigid objects," in *Proc. DARPA Image Understanding Workshop*, 1984.

[20] S. Ullman, "The interpretation of visual motion," Ph. D. thesis, Mass. Inst. of Technol., Dept. of Elect. Eng. Comput. Sci., 1979.

[21] A. Waxman and K. Wohn, "Image flow theory," in *Advances in Computer Vision* (C. M. Brown, Ed.). New York: Erlbaum, 1987.

[22] J. Weng, T. S. Huang, and N. Ahuja, "A two-step approach to optimal motion and structure estimation," in *Proc. IEEE Workshop Comput. Vision*, 1987, pp. 355–357.