Two-view motion analysis: a unified algorithm

Xinhua Zhuang and Thomas S. Huang

Coordinated Science Laboratory, University of Illinois, 1101 W. Springfield Avenue, Urbana, Illinois 61801

Robert M. Haralick

Machine Vision International, 325 E. Eisenhower Parkway, Ann Arbor, Michigan 48104

Received January 29, 1986; accepted May 7, 1986

We present a linear algorithm for determining the three-dimensional rotation and translation of a rigid object from two time-sequential perspective views using point correspondences. The algorithm is different from existing ones in two respects. First, various measures for combating noise are incorporated. Second, the algorithm is unified in the sense that, assuming that the surface assumption holds, it can handle both the case of nonzero translation and the case of zero translation.

1. INTRODUCTION

Determining the relative motion between an observer and his environment is a major problem in computer vision. Its applications include mobile robot navigation and the monitoring of dynamic industrial processes. Motion estimation also has many applications in image processing. For example, in efficient coding using digital pulse code modulation in time, motion estimation and compensation can potentially improve the compression significantly. In reducing noise in image sequences by temporal filtering, registration of the object of interest from frame to frame is necessary to avoid blurring, and registration is, in essence, equivalent to motion estimation. The reader is referred to Refs. 1 and 2 for some of these applications.

In this paper we present an approach to the determination of three-dimensional motion of a single isolated rigid body from two time-sequential perspective views (image frames).

A. Statement of the Problem

The basic geometry of the problem is sketched in Fig. 1. The object-space coordinates are denoted by lowercase letters and the image-space coordinates by uppercase letters. Let the two views be taken at τ_1 and τ_2, respectively, and $\tau_1 < \tau_2$. The coordinates at τ_2 are primed, whereas the coordinates at τ_1 are unprimed. Specifically, consider a particular physical point on the surface of a rigid body in the scene. Let

(x, y, z) = object-space coordinates of the point at time τ_1,

(x', y', z') = object-space coordinates of the point at time τ_2,

(X, Y) = image-space coordinates of the point at time τ_1,

(X', Y') = image-space coordinates of the point at time τ_2.

It is well known in kinematics that

$$(x', y', z')^t = R_o(x, y, z)^t + \mathbf{T}_o, \tag{1}$$

where R_o is a 3×3 orthonormal matrix of the first kind, i.e., $R_o{}^tR_o = R_oR_o{}^t = I_3(I_3 - 3 \times 3$ identity matrix) and $\det(R_o) = 1$, $\mathbf{T}_o = (t_{01}, t_{02}, t_{03})^t$ is a 3×1 vector (column matrix), and t represents the matrix transposition operation.

Our problem is the following:

Given two images frames at τ_1 and τ_2,
Find the motion parameters $\mathbf{T}_o$ (to within a scale factor) and R_0.

As we shall see later, the equations relating the motion parameters to the image-point coordinates inevitably involve the ranges (z coordinates) of the object points. Therefore, in determining the motion parameters, we also determine the ranges of the observed objects points. It will be seen that the translation vector $\mathbf{T}_o$ and the object-point ranges can be determined to within a global positive scale factor. The value of this scale factor could be found if we should know the magnitude of $\mathbf{T}_o$ or the absolute range of any observed object point.

B. A Two-Stage Approach to Solving the Problem

We present a two-stage method to solve the problem posed in Subsection 1.A. In the first stage, we find point correspondences in the two perspective views (images). By a point correspondence, we mean a pair of image coordinates (X_i, Y_i), (X'_i, Y'_i) that are images at τ_1 and τ_2, respectively, of the same physical point on the object. Then, in the second stage, we determine the motion parameters from these image coordinates by solving a set of equations. This paper deals with the second stage. However, a few comments on the first stage are in order here.

In order to be able to find point correspondences, we must have images that contain points that are distinctive in some sense. For example, images of man-made objects often contain sharp corners that are relatively easy to extract.[3] More generally, image points at which the local gray-level variations (defined in some way) are maximum can be used.[4]

In any case, we first extract in each of the two images a large number of points that are distinctive. Then we try to match the two point patterns in the two images by using

0740-3232/86/091492-09$02.00 © 1986 Optical Society of America

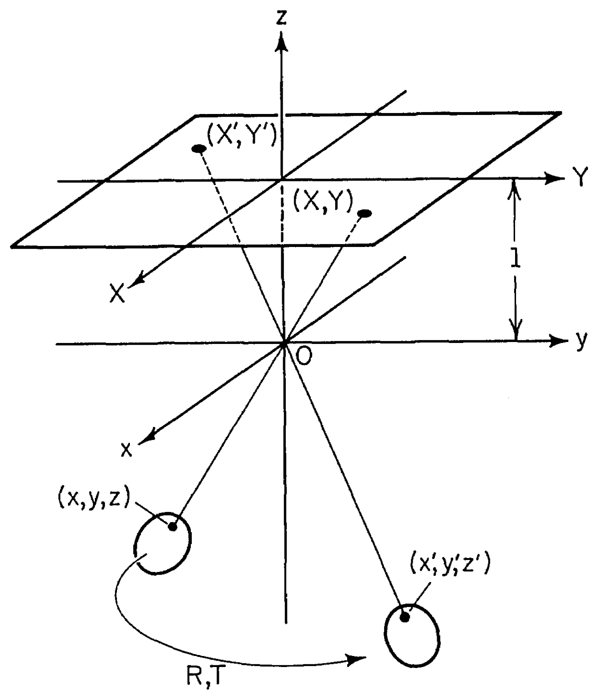

Fig. 1. Basic geometry of the problem.

spatial structures of the patterns.[5] The matching will be successful only if the amount of rotation (θ) is relatively small (so that the perspective distortion is small). For example, in Ref. 5 good matching results were obtained when $\theta < 5$ deg. This restriction may be relaxed if we have some *a priori* information about the object.[6]

C. Motion Equations

From the geometry of Fig. 1 and using Eq. (1), we can derive an equation relating the motion parameters to the coordinates of a corresponding image-point pair. Unfortunately, this equation is nonlinear.[7,8] Iterative techniques for solving nonlinear equations can hardly be expected to converge to the correct solution unless a very good initial guess of the solution is available. Fortunately, by defining appropriate intermediate unknowns, it is possible to put the motion equation into a linear form. However, after these intermediate unknowns are solved for, we have to determine from them the motion parameters. This last step is by no means easy. But at least four algorithms have been proposed for carrying it out (see Refs. 9–13). Longuet-Higgins[14] has also derived a condition under which the linear algorithm will fail.

In the first three algorithms, a tacit assumption is made that the translation $\mathbf{T}_o$ is not zero. They cannot handle the case $\mathbf{T}_o = 0$. In this paper, we shall describe an algorithm that works whether $\mathbf{T}_o$ is nonzero or not.

D. Outline of the Paper

The structure of the paper is as follows. In Section 2, the linear motion equation is derived, and a necessary and sufficient condition is given for its degeneracy. Assuming nondegeneracy, a solution to the linear motion equations is given. At this stage, we have determined the intermediate unknowns. Then, in Section 3, an algorithm is presented for finding the motion parameters from these intermediate variables. There are four candidate solutions. In Section 4, a method is described for using the image-point correspondences to pick out the unique correct solution from the four candidates. Finally, in Section 5, the entire algorithm is summarized.

Our algorithm is different from existing ones. It is probably closest in spirit to the ones given by Zhuang and Haralick[13] and Yen and Huang.[12] The new contributions are twofold. First, the algorithm is unified in the sense that it can handle both the case of $\mathbf{T}_o \neq 0$ and the case of $\mathbf{T}_o = 0$. Second, various measures are taken to combat the effect of noise in image coordinates.

2. TWO-VIEW MOTION EQUATION: GENERAL SOLUTION AND SURFACE ASSUMPTION

A. Derivation of Motion Equation

Referring to Fig. 1, we assume that a rigid body is in motion in the half-space $z < 0$. Take a particular point on the object whose three-dimensional (3-D) spatial coordinates before and after motion are (x, y, z) and (x', y', z'), respectively. Let (X, Y) $[(X', Y')]$ be its central projective coordinates before (after) motion onto the image plane, $z = 1$, with the projective center at the origin O. The following projective equations relate the 3-D spatial coordinates and their corresponding two-dimensional projective coordinates:

$$\begin{cases} X = x/z, & Y = y/z \\ X' = x'/z', & Y' = y'/z' \end{cases}. \tag{2}$$

Recall that the 3-D coordinates of a point before and after the motion are related by Eq. (1). Taking any vector $\mathbf{T}$ that is collinear with $\mathbf{T}_o$ and taking its cross product with both sides of Eq. (1), we obtain

$$\frac{z'}{z}\mathbf{T} \times (X', Y', 1)^t = \mathbf{T} \times [R_o(X, Y, 1)^t] \tag{3}$$

and, after taking dot product of both sides of Eq. (3) with $(X', Y', 1)$,

$$(X', Y', 1)(\mathbf{T} \times R_o)(X, Y, 1)^t = 0, \tag{4}$$

where $\mathbf{T} \times R_o \triangleq [\mathbf{T} \times r_1, \mathbf{T} \times r_2, \mathbf{T} \times r_3]$; r_1, r_2, r_3 being the columns of R_o. Let $E \triangleq \mathbf{T} \times R_o$. Then Eq. (4) states that, for any image-point correspondence pair $[(X, Y), (X', Y')]$, the 3×3 matrix E satisfies the following equation that is linear and homogeneous in the elements of E:

$$(X', Y', 1)E(X, Y, 1)^t = 0. \tag{5}$$

Denote the set of all observed image-point correspondence pairs $(X_i, Y_i) \leftrightarrow (X'_i, Y'_i)$, $i = 1, 2, \ldots, N$, by P. Let

$$A = \begin{bmatrix} X'_1X_1, & X'_1Y_1, & X'_1, & Y'_1X_1, & Y'_1Y_1, & Y'_1, & X_1, & Y_1, & 1 \\ \vdots & & & & & & & \vdots & \\ X'_NX_N, & X'_NY_N, & X'_N, & Y'_NX_N, & Y'_NY_N, & Y'_N, & X_N, & Y_N, & 1 \end{bmatrix},$$

$$E = \begin{bmatrix} h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 \\ h_7 & h_8 & h_9 \end{bmatrix},$$

$$h = (h_1, \ldots, h_9)^t. \tag{6}$$

Then it can be easily seen that the linear equations with $[(X,$

$Y)$, $(X', Y')] \in P$ are equivalent to the following matrix linear equation for h:

$$Ah = 0. \quad (7)$$

Both Eqs. (5) and (7) will be called the two-view motion equations. Our approach is first to solve for the intermediate unknowns h_i and then to obtain the motion parameters from them. Since any $\mathbf{T} \times R_o$ with $\mathbf{T} \times \mathbf{T}_o = 0$ satisfies both Eqs. (5) and (7) (the latter if $\mathbf{T} \times R_o$ is rearranged as h) and, moreover, since such a vector $\mathbf{T}$ that is collinear with $\mathbf{T}_o$ has one degree of freedom when $\mathbf{T}_o \neq 0$ and three degrees of freedom when $\mathbf{T}_o = 0$, the general solution of the two-view motion equation therefore has at least one degree of freedom when $\mathbf{T}_o \neq 0$ and three degrees of freedom when $\mathbf{T}_o = 0$. In other words, the coefficient matrix A in Eq. (7) has a rank no larger than 8 when $\mathbf{T}_o \neq 0$ and no larger than 6 when $\mathbf{T}_o = 0$. If the rank of A equals 8, then the translation $\mathbf{T}_o$ must be nonzero, and the general solution must have one degree of freedom and hence coincides with $\alpha(\mathbf{T}_o \times R_o)$, where α is any real number. If the rank equals 6 and the translation $\mathbf{T}_o$ is zero, then the general solution must have three degrees of freedom and hence coincides with $\mathbf{T}_o \times R_o$, where $\mathbf{T}$ is any real vector.

B. Degeneracy and Surface Assumption

Definition
The two-view motion equation is called degenerate if the rank of A is less than 8 when $\mathbf{T}_o \neq 0$ or less than 6 when $\mathbf{T}_o = 0$. Thus, when the two-view motion equation is not degenerate, any nonzero solution E can be decomposed into $\mathbf{T} \times R_o$ with $\mathbf{T} \times \mathbf{T}_o = 0$. It is apparent that the degeneracy is equivalent to having a nonzero solution E such that E is not equal to $\mathbf{T} \times R_o$ for any $\mathbf{T}$ with $\mathbf{T} \times \mathbf{T}_o = 0$.

However, we have the following.

Lemma
A matrix E is equal to $\mathbf{T} \times R_o$ with $\mathbf{T} \times \mathbf{T}_o = 0$ if and only if $R_o{}^tE + E^tR_o = 0$ and $T_o{}^tE = 0$. (Proof of this lemma is given in Appendix A.)

From this lemma it is clear that the two-view motion equation is degenerate if and only if there is a nonzero solution E such that

$$\|R_o{}^tE + E^tR_o\| + \|T_o{}^tE\| \neq 0. \quad (8)$$

Since the rigid-body motion happens in the half-space ($z < 0$), Eq. (5) is, as is easily seen, equivalent to

$$(x', y', z')E(x, y, z)^t = 0, \quad (9)$$

or, after substituting $[R_o(x, y, z)^t + \mathbf{T}_o]$ for (x', y', z') in Eq. (9),

$$(x, y, z)(R_o{}^tE)(x, y, z)^t + T_o{}^tE)(x, y, z)^t = 0. \quad (10)$$

As a result, the two-view motion equation becomes degenerate if and only if Eq. (10) has a nonzero solution E such that inequality (8) holds when each (x, y, z) in Eq. (10) comes from a group of surface points, S, which is visible before and after motion and produces the set of image-point correspondence pairs, P. Letting $U = R_o{}^tE$, then Eq. (10) for E is equivalent to the following equation for U:

$$(x, y, z)U(x, y, z)^t + T_o{}^tR_oU)(x, y, z)^t = 0. \quad (11)$$

Thus we obtain the following theorems.

Theorem 1
The two-view motion equation is not degenerate if and only if the surface assumption holds, that is, one cannot find a 3×3 matrix U such that the group of surface points S are contained in the following quadratic surface:

$$(x, y, z)U(x, y, z)^t + (T_o{}^tR_oU)(x, y, z)^t = 0, \quad (12)$$

with $\|U + U^t\| + T_o{}^tR_oU\| \neq 0$.

Theorem 2
Under the surface assumption, the two-view motion equation has a rank 8 and a general solution $\alpha\mathbf{T}_o \times R_o$ (where α is any real number) when $\mathbf{T}_o \neq 0$ or a rank 6 and a general solution $\mathbf{T} \times R_o$ (where $\mathbf{T}$ is any real vector) when $\mathbf{T}_o = 0$.

Because of theorem 2, at least six or eight image-point correspondence pairs are needed to ensure the surface assumption, depending on whether the translation $\mathbf{T}_o$ is zero or not. In practice, more pairs are preferable to increase the probability that the surface assumption will be satisfied and to smooth out noise effects. We would like to point out that our surface assumption is equivalent to the condition of Longuet-Higgins[14] when $\mathbf{T}_o \neq 0$. His condition does not include the case $\mathbf{T}_o = 0$.

The surface assumption, as stated in theorem 1, has the following interpretation. To simplify the interpretation, we assume that the object is stationary and that the camera is moving. Let the origin of the cameras system be 0 and 0′, respectively, before and after the motion. Then, for $\mathbf{T}_o \neq 0$, the surface assumption holds if and only if the 3-D points corresponding to P do not lie on a quadratic surface passing through 0 and 0′. For $\mathbf{T}_o = 0$ (then 0 and 0′ coincide), the surface assumption holds if and only if the 3-D points corresponding to P do not lie on a cone with its apex at 0.

C. Solving the Motion Equations

Now we come to the question of computing E or h. There are a number of possibilities. We propose the following procedure. From eight or more point correspondences $(X_i, Y_i) \leftrightarrow (X'_i, Y'_i)$, $i = 1, 2 \ldots, N$ $(N \geq 8)$, we form the positive semidefinite and symmetrical matrix $W = A^tA$, where A is given by Eq. (6). Then we find h to minimize h^tWh under the constraint $\|h\| = 1$. The solution is the eigenvector of W associated with the smallest eigenvalue. The motivation for this method is as follows. In the absence of noise, we have

$$Ah = 0. \quad (7)$$

When noise is present in the image-point coordinates, Eq. (7) is no longer valid. A reasonable thing to do is to find a least-squares solution, i.e., to minimize

$$\|Ah\|^2 = (Ah)^t(Ah) = h^tA^tAh = h^tWh.$$

Assume that the surface assumption holds. Then, if $\mathbf{T}_o \neq 0$, the rank of W is 8 in the absence of noise. One and only one eigenvalue of W will be zero, and the corresponding eigenvector gives us the exact solution. In the presence of noise (which is assumed to be small), the smallest eigenvalue will be almost zero, and the corresponding eigenvector gives us a least-squares solution. When $\mathbf{T}_o = 0$, the rank of W is 6 in the absence of noise. Three eigenvalues will be zero, and the corresponding eigenvector (after normalization to unit magnitude) will have two degrees of freedom. In the presence of (small) noise, three of the eigenvalues of W will be almost zero. A linear combination of the three corresponding eigenvectors is our least-squares solution.

In any real problem, the probability that the surface assumption is violated is extremely low. In the remainder of this paper we shall assume that the surface assumption is always valid.

3. DECOMPOSING E

Under the surface assumption, any nonzero solution E of the two-view motion equation has a decomposition $\mathbf{T} \times R_o$ with $\mathbf{T} \times \mathbf{T}_o = 0$. In Ref. 11 it is proved that, if a nonzero matrix E has a decomposition $\mathbf{T} \times R$, with R being an orthonormal matrix of the first kind, E will admit one and only one alternative decomposition $(-\mathbf{T}) \times R'$, with R' being an orthonormal matrix of the first kind. Thus the surface assumption ensures that any nonzero solution E of the two-view motion equation admits two and only two decompositions:

$$E = \mathbf{T} \times R = (-\mathbf{T}) \times R', \tag{13}$$

where either R or R' equals R_o and $\mathbf{T} \times \mathbf{T}_o = 0$. In what follows we derive a procedure to decompose E. In this procedure, measures are taken to combat the effect of noise.

A. Determining Translation

Let

$$\mathbf{T} = (t_1, t_2, t_3)^t$$

$$R = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}, \qquad R' = \begin{bmatrix} R'_1 \\ R'_2 \\ R'_3 \end{bmatrix}, \tag{14}$$

$$E = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix}$$

where R_i, R'_i, E_i are all 1×3 row matrices. Then it is easy to see, by means of Eq. (13), that

$$E_1 = t_2R_3 - t_3R_2 \tag{15}$$

$$= -t_2R'_3 + t_3R'_2, \tag{15'}$$

$$E_2 = t_3R_1 - t_1R_3 \tag{16}$$

$$= -t_3R'_1 + t_1R'_3, \tag{16'}$$

$$E_3 = t_1R_2 - t_2R_1 \tag{17}$$

$$= -t_1R'_2 + t_2R'_1. \tag{17'}$$

Thus $\mathbf{T}$ can be determined up to a sign by solving the following equations:

$$\|E_1\|^2 = t_2^2 + t_3^2, \quad \|E_2\|^2 = t_3^2 + t_1^2, \quad \|E_3\|^2 = t_1^2 + t_3^2, \tag{18}$$

$$(E_1, E_2) = t_1t_2, \tag{19}$$

$$(E_2, E_3) = t_2t_3, \tag{20}$$

$$(E_3, E_1) = t_3t_1 \tag{21}$$

$[(E_i, E_j)$ denotes the dot product of E_i and $E_j]$,

where, as is easily seen, Eqs. (18) are equivalent to Eqs. (22)–(24):

$$\|E_2\|^2 + \|E_3\|^2 - \|E_1\|^2 = 2t_1^2, \tag{22}$$

$$\|E_3\|^2 + \|E_1\|^2 - \|E_2\|^2 = 2t_2^2, \tag{23}$$

$$\|E_1\|^2 + \|E_2\|^2 - \|E_3\|^2 = 2t_3^2. \tag{24}$$

From noise considerations, we recommend a scheme to compute $\pm\mathbf{T}$ as follows:

Step 1. If $|t_1| \geqslant |t_2|, |t_3|$ in Eqs. (22)–(24), then $\pm\mathbf{T}$ are determined by using Eqs. (22), (19), and (21). Stop.

Step 2. If $|t_2| \geqslant |t_3|$ in Eqs. (23) and (24), then $\pm\mathbf{T}$ are determined by using Eqs. (23), (19), and (20). Stop.

Step 3. $\pm\mathbf{T}$ are determined by using Eqs. (24), (20), and (21). Stop.

B. Determining Rotation

Once $\pm\mathbf{T}$ are determined, R and R' could be computed by means of Eqs. (15)–(17) and (15′)–(17′). In fact, a simple manipulation leads to

$$\begin{aligned} E_1 \times E_2 &= t_3(t_1R_1 + t_2R_2 + t_3R_3), \\ E_2 \times E_3 &= t_1(t_1R_1 + t_2R_2 + t_3R_3), \\ E_3 \times E_1 &= t_2(t_1R_1 + t_2R_2 + t_3R_3), \end{aligned} \tag{25}$$

$$\begin{aligned} (E_1 \times E_2) \times E_3 &= t_3[(t_1^2 + t_2^2)R_3 - t_3(t_1R_1 + t_2R_2)], \\ (E_2 \times E_3) \times E_1 &= t_1[(t_2^2 + t_3^2)R_1 - t_1(t_2R_2 + t_3R_3)], \\ (E_3 \times E_1) \times E_2 &= t_2[(t_3^2 + t_1^2)R_2 - t_2(t_3R_3 + t_1R_1)], \end{aligned} \tag{26}$$

and hence

$$t_1\|\mathbf{T}\|^2R_1 = (E_2 \times E_3) \times E_1 + t_1(E_2 \times E_3), \tag{27}$$

$$t_2\|\mathbf{T}\|^2R_1 = (E_3 \times E_1) \times E_2 + t_2(E_3 \times E_1), \tag{28}$$

$$t_3\|\mathbf{T}\|^2R_3 = (E_1 \times E_2) \times E_3 + t_3(E_1 \times E_2), \tag{29}$$

which, combined with Eqs. (15)–(17), determine R. For instance, when $|t_1| \geqslant |t_2|, |t_3|$, we use Eqs. (27), (17), and (16) to compute R_1, R_2, and R_3, respectively, and so on.

Similarly, we could obtain

$$-t_1\|\mathbf{T}\|^2R'_1 = (E_2 \times E_3) \times E_1 - t_1(E_2 \times E_3), \tag{27'}$$

$$-t_2\|\mathbf{T}\|^2R'_1 = (E_3 \times E_1) \times E_2 - t_2(E_3 \times E_1), \tag{28'}$$

$$-t_3\|\mathbf{T}\|^2R'_3 = (E_1 \times E_2) \times E_3 - t_3(E_1 \times E_2), \tag{29'}$$

which, combined with Eqs. (15′)–(17′), determine R'.

Thus we have outlined a direct procedure to compute $\mathbf{T}, R, R'$ from E. In the next section we discuss how to determine the true rotation, the true translation direction, and the relative ranges of observed points from T, R, R'. Here we would like to point out what happens with the decompositions when noise is present in the measurements of image-point coordinates. In general, an erroneous nonzero solution $\bar{E}$ does not admit any decompositions, as in Eq. (13). However, by using the above procedure, we still can compute a vector $\bar{\mathbf{T}}$ and two matrices $\bar{R}$ and $\bar{R}'$. The triplet $(\bar{\mathbf{T}}, \bar{R}, \bar{R}')$

should approach the true triplet $(\mathbf{T}, R, R')$ when the noise tends to zero. In other words, $(\bar{\mathbf{T}}, \bar{R}, \bar{R}')$ should be closer to $(\mathbf{T}, R, R')$ when the noise becomes smaller. The two matrices $\bar{R}$ and $\bar{R}'$ might not be orthonormal matrices of the first kind. However, algorithms exist for constructing two orthonormal matrices of the first kind, $\tilde{R}$ and $\tilde{R}'$, which are approximations of $\bar{R}$ and $\bar{R}'$ (and hence R and R'), respectively; see, for example, Arun *et al.*[15]

4. DETERMINING THREE-DIMENSIONAL MOTION PARAMETERS AND SURFACE STRUCTURE FROM T, R

A. Determining Rotation and Translation Direction

Under the surface assumption, any nonzero solution E admits two and only two decompositions, as in Eq. (13). The next task is to determine the true rotation from R and R', the true translation direction from $\pm\mathbf{T}$, and also the relative depths $z/\|\mathbf{T}_o\|$ and $z'\|\mathbf{T}_o\|$ when $\mathbf{T}_o \neq 0$. What we really need is a criterion function $L(\cdot, \cdot)$, where the first argument is a 3×1 vector and the second a 3×3 matrix such that $L(\mathbf{T}, R)$ equals zero if and only if R equals R_o and $\mathbf{T}$ has the same direction as $\mathbf{T}_o$. Note that a zero vector has an indefinite direction; in other words, it has the same direction as any other vector. Thus such a function L should satisfy the following conditions:

$$\begin{cases} L(\mathbf{T}, R) = 0, \quad L(-\mathbf{T}, R) \neq 0, \quad L(\pm\mathbf{T}, R') \neq 0 \\ \text{if } R = R_o, \quad \mathbf{T}_o \neq 0, \quad \text{and } \mathbf{T}/\|\mathbf{T}\| = \mathbf{T}_o/\|\mathbf{T}_o\| \end{cases}; \tag{30}$$

$$\begin{cases} L(\pm\mathbf{T}, R) = 0, \quad L(\pm\mathbf{T}, R') \neq 0 \\ \text{if } R = R_o, \quad T_o = 0 \end{cases}. \tag{31}$$

To see how we should design such a criterion function, we return to the source information, the 3-D rigid-body motion equation (1). Suppose that R equals R_o and $\mathbf{T}$ has the same direction as $\mathbf{T}_o$. If $\mathbf{T}_o \neq 0$, then there must be a constant $\alpha > 0$ such that $\mathbf{T} = \alpha\mathbf{T}_o$. Thus it follows from Eqs. (1) and (2) that

$$\alpha z'(X', Y', 1)^t = \alpha z R(X, Y, 1)^t + \mathbf{T}; \tag{32}$$

hence

$$\begin{aligned} \alpha z'(X', Y', 1)^t \times [R(X, Y, 1)^t] &= \mathbf{T} \times [R(X, Y, 1)^t], \\ \alpha z(X', Y', 1)^t \times [R(X, Y, 1)^t] &= \mathbf{T} \times (X', Y', 1)^t. \end{aligned} \tag{33}$$

For abbreviation, we let

$$\begin{aligned} \mathbf{v} &= (X, Y, 1)^t, \\ \mathbf{v}' &= (X', Y', 1)^t. \end{aligned} \tag{34}$$

Since $\alpha > 0$, $z < 0$, $z' < 0$, from Eqs. (33) we obtain

$$\begin{aligned} \alpha z'\|\mathbf{v}' \times \mathbf{Rv}\| &= -\|\mathbf{T} \times \mathbf{Rv}\|, \\ \alpha z\|\mathbf{v}' \times \mathbf{Rv}\| &= -\|\mathbf{T} \times \mathbf{v}'\|. \end{aligned} \tag{35}$$

Then, multiplying both sides of Eq. (32) by $\|\mathbf{v}' \times \mathbf{Rv}\|$, and substituting $-\|\mathbf{T} \times \mathbf{Rv}\|$ for $\alpha z'\|\mathbf{v}' \times \mathbf{Rv}\|$ and $-\|\mathbf{T} \times \mathbf{v}'\|$ for $\alpha z\|\mathbf{v}' \times \mathbf{Rv}\|$ because of Eqs. (35), we have

$$-\|\mathbf{T} \times \mathbf{Rv}\|\mathbf{v}' = -\|\mathbf{T} \times \mathbf{v}'\|\mathbf{Rv} + \|\mathbf{v}' \times \mathbf{Rv}\|\mathbf{T}, \tag{36}$$

or, after rearrangement,

$$\|\mathbf{T} \times \mathbf{Rv}\|\mathbf{v}' - \|\mathbf{T} \times \mathbf{v}'\|\mathbf{Rv} + \|\mathbf{v}' \times \mathbf{Rv}\|\mathbf{T} = 0. \tag{37}$$

Denoting the left-hand side of Eq. (37) by $H(\mathbf{v}, \mathbf{v}', \mathbf{T}, R)$, we conclude that for each pair $(\mathbf{v}, \mathbf{v}') \in P$ the function $H(\mathbf{v}, \mathbf{v}', \mathbf{T}, R)$ equals zero whenever R equals R_o and $\mathbf{T} = \alpha\mathbf{T}_o$, $\alpha > 0$ (assuming that $\mathbf{T}_o \neq 0$). If $\mathbf{T}_o = 0$, then both $H(\mathbf{v}, \mathbf{v}', \mathbf{T}, R)$ and $H(\mathbf{v}, \mathbf{v}', -\mathbf{T}, R)$ equal zero whenever R equals R_o, since in this case $\mathbf{v}'$ has the same direction as $\mathbf{Rv}$ and hence $\mathbf{v}' \times \mathbf{Rv} = 0$ and $\|\mathbf{T} \times \mathbf{Rv}\|\mathbf{v}' = \|\mathbf{T} \times \mathbf{v}'\|\mathbf{Rv}$. Thus, letting

$$L(\mathbf{T}, R) = \sum_{(\mathbf{v},\mathbf{v}') \in P} \|H(\mathbf{v}, \mathbf{v}', \mathbf{T}, R)\|/(\|\mathbf{v}\| \cdot \|\mathbf{v}'\| \cdot \|\mathbf{T}\|), \tag{38}$$

we have proved that $L(\mathbf{T}, R) = 0$ whenever R equals R_o and $\mathbf{T}$ has the same direction as $\mathbf{T}_o$. In what follows we verify that the function $L(\mathbf{T}, R)$ is just what we want. For this, the only thing that we need to verify is

$$\begin{cases} L(-\mathbf{T}, R) > 0, \quad L(\pm\mathbf{T}, R') > 0 \\ \text{if } R = R_o, \quad \mathbf{T}_o \neq 0, \quad \text{and } \mathbf{T}_o/\|\mathbf{T}_o\| = \mathbf{T}/\|\mathbf{T}\| \end{cases} \tag{39}$$

and

$$\begin{cases} L(\pm\mathbf{T}, R') > 0 \\ \text{if } R = R_o, \quad \mathbf{T}_o = 0 \end{cases}. \tag{40}$$

We would like to point out that the main purpose of the normalization and summation in Eq. (38) is to smooth out noise effects.

To provide expressions (39) and (40), we need to derive an explicit relation between R and R'. From Eqs. (15)–(17) it follows that

$$t_3(R_2 + R'_2) = t_2(R_3 + R'_3), \tag{41}$$

$$t_1(R_3 + R'_3) = t_3(R_1 + R'_1), \tag{42}$$

$$t_2(R_1 + R'_1) = t_1(R_2 + R'_2), \tag{43}$$

and hence

$$\begin{aligned} R + R' &= \mathbf{T}(R_1 + R'_1)/t_1 \qquad \text{when } t_1 \neq 0, &(44) \\ &= \mathbf{T}(R_2 + R'_2)/t_2 \qquad \text{when } t_2 \neq 0, &(45) \\ &= \mathbf{T}(R_3 + R'_3)/t_3 \qquad \text{when } t_3 \neq 0. &(46) \end{aligned}$$

In any case, there exists a row vector $\mathbf{q}$ such that

$$R + R' = \mathbf{T} \cdot \mathbf{q}. \tag{47}$$

Also, we need the following simple fact: Except for at most one pair $(\mathbf{v}, \mathbf{v}')$ in P, the following inequalities hold:

$$\mathbf{T} \times \mathbf{v}' \neq 0, \tag{48}$$

$$\mathbf{T} \times \mathbf{Rv} \neq 0. \tag{49}$$

In fact, it is obvious that, except for at most one pair, inequality (48) holds. Then, when $\mathbf{T}_o \neq 0$, expressions (35) and (48) imply inequality (49); and when $\mathbf{T}_o = 0$, the nonzero vector $\mathbf{Rv}$ has the same direction as $\mathbf{v}'$ and hence inequality (48) also implies inequality (49).

Now we are ready to prove expressions (39) and (40). We need to prove that $L(\pm\mathbf{T}, R') > 0$, in general, and $L(-\mathbf{T}, R) > 0$ when $\mathbf{T}_o \neq 0$.

$L(\pm\mathbf{T}, R') > 0$: We need only to prove that, for a pair $(\mathbf{v}, \mathbf{v}')$ that satisfies inequalities (48) and (49), the function $H(\mathbf{v}, \mathbf{v}', \pm\mathbf{T}, R')$ is not equal to zero. As a matter of fact, an even stronger result exists: For any positive numbers, λ' and λ, the following inequality holds:

$$\lambda'\mathbf{v}' - \lambda R'\mathbf{v} \pm \|\mathbf{v}' \times R'\mathbf{v}\|\mathbf{T} \neq 0. \tag{50}$$

Using relation (47) and the motion equation (1), we could rewrite $\lambda'\mathbf{v}' - \lambda R'\mathbf{v}$ as

$$\begin{aligned}\lambda'\mathbf{v}' - \lambda R'\mathbf{v} &= \lambda'\mathbf{v}' - \lambda(-R + \mathbf{T}\cdot\mathbf{q})\mathbf{v}\\ &= \lambda'\mathbf{v}' + \lambda\mathbf{R}\mathbf{v} - \lambda\mathbf{T}(\mathbf{q}\mathbf{v})\\ &= \lambda'\mathbf{v}' + \frac{\lambda}{z}(z'\mathbf{v}' - \mathbf{T}_o) - \lambda\mathbf{T}(\mathbf{q}\mathbf{v})\\ &= \left(\lambda' + \frac{\lambda z'}{z}\right)\mathbf{v}' - \frac{\lambda}{z}\mathbf{T}_o - \lambda(\mathbf{q}\mathbf{v})\mathbf{T}.\end{aligned} \tag{51}$$

Thus the left-hand side of inequality (50) consists of two terms

$$\left(\lambda' + \frac{\lambda z'}{z}\right)\mathbf{v}' \tag{52}$$

and

$$-\frac{\lambda}{z}\mathbf{T}_o - \lambda(\mathbf{q}\mathbf{v})\mathbf{T} \pm \|\mathbf{v}' \times R'\mathbf{v}\|\mathbf{T}. \tag{53}$$

The coefficient of $\mathbf{v}'$ in the first term (52) is positive. The second term (53) is collinear with $\mathbf{T}$. And the two vectors, $\mathbf{v}'$ and $\mathbf{T}$, are not collinear with each other because of inequality (48). Thus their sum, the left-hand side of inequality (50), cannot be zero. This completes the proof of $L(\pm\mathbf{T}, R') > 0$.

$L(-\mathbf{T}, R) > 0$ when $\mathbf{T}_o \neq 0$: This is simple since, except for at most one pair $(\mathbf{v}, \mathbf{v}')$, $\mathbf{v}' \times \mathbf{R}\mathbf{v}$ does not equal zero when $\mathbf{T}_o \neq 0$ because of expressions (35) and (48), and hence $H(\mathbf{v}, \mathbf{v}', -\mathbf{T}, R)$ cannot be zero:

$$\begin{aligned}H(\mathbf{v}, \mathbf{v}', -\mathbf{T}, R) &= H(\mathbf{v}, \mathbf{v}', \mathbf{T}, R) - 2\|\mathbf{v}' \times \mathbf{R}\mathbf{v}\|\mathbf{T}\\ &= -2\|\mathbf{v}' \times \mathbf{R}\mathbf{v}\|\mathbf{T} \neq 0.\end{aligned}$$

So, finally, we have proved the theorem given below.

Theorem 3

Assume that the surface assumption holds and that E $(= \mathbf{T} \times R = (-\mathbf{T}) \times R')$ is a nonzero solution of the two-view motion equation. Then $R = R_o$ and $\mathbf{T}$ has the same directions as $\mathbf{T}_o$ if and only if

$$L(\mathbf{T}, R) = 0. \tag{54}$$

B. Determining Relative Depths

Now it is easy to prove the following.

Theorem 4

Assume that R equals R_o and $\mathbf{T}$ has the same direction as $\mathbf{T}_o$. Then, when $\mathbf{T}_o \neq 0$, the relative depths are given by

$$\begin{aligned}z/\|\mathbf{T}_o\| &= -\frac{\|\mathbf{T} \times \mathbf{v}'\|}{\|T\| \cdot \|\mathbf{v}' \times R_o\mathbf{v}\|},\\ z'/\|\mathbf{T}_o\| &= -\frac{\|\mathbf{T} \times R_o\mathbf{v}\|}{\|\mathbf{T}\| \cdot \|\mathbf{v}' \times R_o\mathbf{v}\|}.\end{aligned} \tag{55}$$

Proof. From the assumptions, Eqs. (35) (with $\alpha = \|\mathbf{T}\|/\|\mathbf{T}_o\|$) hold. Thus Eqs. (55) immediately follow, where the minuses are due to $z < 0$, $z' < 0$.

It is easy to argue that, except at most one point correspondence pair, $\mathbf{T} \times \mathbf{v}'$ is nonzero. Thus, except at most one pair, $\mathbf{v}' \times R_o\mathbf{v}$ is nonzero by Eqs. (35). This indicates that the division in Eqs. (55) is no trouble.

Q.E.D. Theorems 3 and 4 indicate that the rotation, the translation direction, and the relative depth map can all be determined under the surface assumption without knowing the mode of the motion, i.e., irrespective of whether the translation is zero.

C. Noise Effects

For convenience, we could modify theorem 3 as follows: R equals R_0 and $\mathbf{T}$ has the same direction as $\mathbf{T}_o$ if and only if

$$\min[L(\mathbf{T}, R), L(-\mathbf{T}, R)] < \min[L(\mathbf{T}, R'), L(-\mathbf{T}, R')], \tag{56}$$

$$L(\mathbf{T}, R) \leqslant L(-\mathbf{T}, R), \tag{57}$$

where relation (56) is used to determine the true rotation and after that relation (57) is used to determine the true translation direction. The equal sign in relation (57) is possible only when $\mathbf{T}_o = 0$.

When noise appears in the measurements, the triplet $(\mathbf{T}, R, R')$ cannot be accurately computed. However, if the noise is small, the computed triplet $(\bar{\mathbf{T}}, \bar{R}, \bar{R}')$ will be close to $(\mathbf{T}, R, R')$, and, as is easily seen, $L(\pm\bar{\mathbf{T}}, \bar{R})$ and $L(\pm\bar{\mathbf{T}}, \bar{R}')$ will also be close to $L(\pm\mathbf{T}, R)$ and $L(\pm\mathbf{T}, R')$, respectively. Therefore relation (56) will imply that

$$\min[L(\bar{\mathbf{T}}, \bar{R}), L(-\bar{\mathbf{T}}, \bar{R})] < \min[L(\bar{\mathbf{T}}, \bar{R}'), L(-\bar{\mathbf{T}}, \bar{R}')], \tag{56'}$$

and relation (57) when $\mathbf{T}_o \neq 0$ will imply that

$$L(\bar{\mathbf{T}}, \bar{R}) < L(-\bar{\mathbf{T}}, \bar{R}).$$

As a result, relation (56′) and the following relation (57′) should give correct approximations of the rotation and the translation direction:

$$L(\bar{\mathbf{T}}, \bar{R}) \leqslant L(-\bar{\mathbf{T}}, \bar{R}). \tag{57'}$$

5. SUMMARY OF THE ALGORITHM

Now we are ready to give the following unified algorithm that does not require the mode of motion to be known.

Step 1

Find h to minimize h^tWh under the constraint $\|h\| = 1$. (If the solution is not unique, pick any solution.)

Step 2

Let

$$E_1 = (h_1, h_2, h_3),$$
$$E_2 = (h_4, h_5, h_6),$$
$$E_3 = \begin{bmatrix} E_1 \\ E_2 \\ E_3 \end{bmatrix}.$$

Step 3

$$a = (\|E_2\|^2 + \|E_3\|^2 - \|E_1\|^2)/2,$$
$$b = (\|E_3\|^2 + \|E_1\|^2 - \|E_2\|^2)/2,$$
$$c = (\|E_1\|^2 + \|E_2\|^2 - \|E_3\|^2)/2.$$

Step 4
If ($a \geq b, c$) then let

$$\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} \sqrt{a} \\ -(E_1, E_2)/\sqrt{a} \\ -(E_1, E_3)/\sqrt{a} \end{bmatrix},$$

$$R_1 = [(E_2 \times E_3) \times E_1 + t_1(E_2 \times E_3)]/(t_1\|\mathbf{T}\|^2),$$

$$R'_1 = [(E_2 \times E_3) \times E_1 - t_1(E_2 \times E_3)]/(-t_1\|\mathbf{T}\|^2),$$

$$R_2 = (E_3 + t_2R_1)t_1,$$

$$R'_2 = (E_3 - t_2R'_1)/(-t_1),$$

$$R_3 = (-E_2 + t_3R_1)t_1,$$

$$R'_3 = (-E_2 - t_3R'_1)/(-t_1)$$

and GO TO Step 7.

Step 5
If ($b \geq c$), then let

$$\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} -(E_2, E_1)/\sqrt{b} \\ \sqrt{b} \\ -(E_2, E_3)/\sqrt{b} \end{bmatrix},$$

$$R_2 = [(E_3 \times E_1) \times E_2 + t_2(E_3 \times E_1)]/(t_2\|\mathbf{T}\|^2),$$

$$R'_2 = [(E_3 \times E_1) \times E_2 - t_2(E_3 \times E_1)]/(-t_2\|\mathbf{T}\|^2),$$

$$R_3 = (E_1 + t_3R_2)/t_2,$$

$$R'_3 = (E_1 - t_3R'_2)/(-t_2),$$

$$R_1 = (-E_3 + t_1R_2)/t_2,$$

$$R'_1 = (-E_3 - t_1R'_2)/(-t_2)$$

and GO TO Step 7.

Step 6
Let

$$\mathbf{T} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} = \begin{bmatrix} -(E_3, E_t)/\sqrt{c} \\ -(E_3, E_2)/\sqrt{c} \\ \sqrt{c} \end{bmatrix},$$

$$R_3 = [(E_1 \times E_2) \times E_3 + t_3(E_1 \times E_2)]/(t_3\|\mathbf{T}\|^2),$$

$$R'_3 = [(E_1 \times E_2) \times E_3 - t_3(E_1 \times E_2)]/(-t_3\|\mathbf{T}\|^2),$$

$$R_1 = (E_2 + t_1R_3)/t_3,$$

$$R'_1 = (E_2 - t_1R'_3)/(-t_3),$$

$$R_2 = (-E_1 + t_2R_3)/t_3,$$

$$R'_2 = (-E_1 - t_2R'_3)/(-t_3).$$

Step 7
Let

$$R = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix},$$

$$R' = \begin{bmatrix} R'_1 \\ R'_2 \\ R'_3 \end{bmatrix}.$$

Step 8
If

$$\min[L(\mathbf{T}, R), L(-\mathbf{T}, R)] < \min[L(\mathbf{T}, R'), L(-\mathbf{T}, R')],$$

then

$$R_o = R.$$

Otherwise

$$R_o = R'.$$

Step 9
If

$$L(\mathbf{T}, R_o) \leq L(-\mathbf{T}, R_o),$$

then $\mathbf{T}_o$ has the same direction as $\mathbf{T}$. Otherwise, $\mathbf{T}_o$ has the same direction as ($-\mathbf{T}$).

Step 10
When $\mathbf{T}_o \neq 0$, the relative depths are given by

$$\frac{z}{\|\mathbf{T}_o\|} = -\frac{\|\mathbf{T}x\mathbf{v}'\|}{\|\mathbf{T}\| \cdot \|\mathbf{v}'xR_o\mathbf{v}\|}, \qquad \frac{z'}{\|\mathbf{T}_o\|} = -\frac{\|\mathbf{T}xR_o\mathbf{v}\|}{\|\mathbf{T}\| \cdot \|\mathbf{v}'xR_o\mathbf{v}\|}.$$

Step 11.
STOP.

Simulation 1

$$\mathbf{T}_o = (0, 0, 0)^t, \qquad R_o = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ -1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Six points in image plane $z = 1$ before motion:

(0.63, −0.93),
(2.09, 0.10),
(0.53, 1.43),
(1.85, 1.83),
(1.29, 0.41),
(−1.32, −0.12).

Six points in image plane $z = 1$ after motion:

(−0.21, −1.10),
(1.54, −1.41),
(1.39, 0.63),
(2.60, −0.01),
(1.20, −0.62),
(−1.01, 0.85),

Computed $E, R, R', \mathbf{T}$:

$$E = \begin{bmatrix} 0.27 & -0.27 & -0.02 \\ 0.27 & 0.27 & -0.59 \\ -0.41 & 0.43 & -0.00 \end{bmatrix},$$

$$R = \begin{bmatrix} 0.71 & 0.71 & 0.00 \\ -0.71 & 0.71 & -0.00 \\ 0.00 & -0.00 & 1.00 \end{bmatrix},$$

$$R' = \begin{bmatrix} 0.32 & 0.25 & 0.91 \\ 0.67 & -0.74 & -0.03 \\ 0.67 & 0.63 & -0.40 \end{bmatrix},$$

$$\mathbf{T} = \begin{bmatrix} 0.59 \\ -0.02 \\ 0.39 \end{bmatrix}.$$

$$L(\mathbf{T}, R) = 0.00.$$

$$L(-\mathbf{T}, R) = 0.00.$$

$$L(\mathbf{T}, R') = 6.95.$$

$$L(-\mathbf{T}, R') = 6.95.$$

$$\min[L(\mathbf{T}, R), L(-\mathbf{T}, R)] = 0.00.$$

$$\min[L(\mathbf{T}, R'), L(-\mathbf{T}, R')] = 6.95.$$

Thus the algorithm gives the correct rotation $R_o = R$.

Simulation 2

$$\mathbf{T}_o = (0, 0, -1)^t, \qquad R_o = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} & 0 \\ -1/\sqrt{2} & 1/\sqrt{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Eight points in image plane $z = 1$ before motion:

$(-0.04, 0.96)$,
$(-0.09, -1.22)$,
$(-0.67, 0.91)$,
$(1.17, 1.29)$,
$(1.10, 0.65)$,
$(-0.13, -0.98)$,
$(-1.13, -1.19)$,
$(1.03, -0.37)$.

Eight points in image plane $z = 1$ after motion:

$(0.41, 0.44)$,
$(-0.60, -0.52)$,
$(0.10, 0.67)$,
$(1.07, 0.06)$,
$(0.62, -0.16)$,
$(-0.45, -0.35)$,
$(-0.89, -0.02)$,
$(0.29, -0.62)$.

Computed $E, R, R', \mathbf{T}$:

$$E = \begin{bmatrix} -0.50 & 0.50 & 0.00 \\ -0.50 & -0.50 & -0.00 \\ -0.00 & 0.00 & 0.00 \end{bmatrix},$$

$$R = \begin{bmatrix} -0.71 & -0.71 & -0.00 \\ 0.71 & -0.71 & -0.00 \\ -0.00 & -0.00 & 1.00 \end{bmatrix},$$

$$R' = \begin{bmatrix} 0.71 & 0.71 & 0.00 \\ -0.71 & 0.71 & 0.00 \\ -0.00 & 0.00 & 1.00 \end{bmatrix},$$

$$\mathbf{T} = \begin{bmatrix} -0.00 \\ 0.00 \\ 0.71 \end{bmatrix}.$$

$$L(\mathbf{T}, R) = 12.33.$$

$$L(-\mathbf{T}, R) = 9.40.$$

$$L(\mathbf{T}, R') = 4.03.$$

$$L(-\mathbf{T}, R') = 0.00.$$

$$\min[L(\mathbf{T}, R), L(-\mathbf{T}, R)] = 9.40.$$

$$\min[L(\mathbf{T}, R'), L(-\mathbf{T}, R')] = 0.00.$$

Thus the algorithm gives the correct rotation, $R_o = R'$, and the correct translation orientation, $\mathbf{T}_o/\|\mathbf{T}_o\| = -\mathbf{T}/\|\mathbf{T}\|$. The latter is due to $L(-\mathbf{T}, R') < L(\mathbf{T}, R')$.

APPENDIX A

Lemma
$E = \mathbf{T} \times R_o$ with $\mathbf{T} \times \mathbf{T}_o = 0$ if and only if $R_o{}^tE + E^tR_o = 0$ and $T_0{}^t\mathrm{E} = 0$.

Proof

Only-If Part
Assume that $E = \mathbf{T} \times R_o$ with $\mathbf{T} \times \mathbf{T}_o = 0$. Then it follows that

$$E = [\mathbf{T} \times r_1, \mathbf{T} \times r_2, \mathbf{T} \times r_3],$$

$$R_o{}^tE = [(r_i, \mathbf{T} \times r_j)],$$

$$E^tR_o = [(\mathbf{T} \times r_i, r_j)],$$

$$T_o{}^tE = [(\mathbf{T}_o, \mathbf{T} \times r_1), (\mathbf{T}_o, \mathbf{T} \times r_2), (\mathbf{T}_o, \mathbf{T} \times r_3)],$$

where $R_o \triangleq [r_1, r_2, r_3]$. And hence we obtain

$$R_o{}^tE + E^tR_o = 0 \qquad \text{by } (r_i, \mathbf{T} \times r_j) + (\mathbf{T} \times r_1, r_j) = 0,$$

$$T_o{}^tE = 0 \qquad \text{by } (\mathbf{T}_o, \mathbf{T} \times r_i) = 0.$$

If Part
Assume that $R_o{}^tE + E^tR_o = 0$ and $T_o{}^tE = 0$. Let $G = ER_o{}^t$. Then it is easy to see that

$$G^t = R_oE^t = R_o(-R_o{}^tER_o{}^t) = -ER_o{}^t,$$

which indicates that G is a skew-symmetrical matrix. Thus there is a $\mathbf{T}$ (see Yen and Huang[12]) so that

$$E = GR_o = \mathbf{T} \times R_o$$

and hence

$$(T_o \times \mathbf{T})^t = \mathbf{T}_o{}^t G = T_o{}^t E R_o{}^t = 0,$$

which completes the "if" part. Q.E.D.

ACKNOWLEDGMENT

This research was supported by the National Science Foundation under grant DMC-8319509.

REFERENCES

1. T. S. Huang, ed., *Image Sequence Analysis* (Springer-Verlag, Berlin, 1981).
2. T. S. Huang, ed., *Image Sequence Processing and Dynamic Scene Analysis* (Springer-Verlag, Berlin, 1983).
3. J. Q. Fang and T. S. Huang, "A corner finding algorithm for image analysis and registration," presented at the Annual Conference of the American Association for Artificial Intelligence, Pittsburgh, Pa., 1982.
4. H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover," Ph.D. dissertation (Stanford University, Stanford, Calif., September 1980).
5. J. Q. Fang and T. S. Huang, "Some experiments on estimating the 3-D motion parameters of a rigid body from two consecutive image frames," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-6,** 547–554 (1984).
6. W. K. Gu, J. Y. Yang, and T. S. Huang, "Matching perspective views of a 3-D object using composite circuits," presented at the Seventh International Conference on Pattern Recognition, Montreal, Canada, 1984.
7. J. W. Roach and J. K. Aggarwal, "Determining the movement of objects from a sequence of images," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-6,** 554–562 (1980).
8. T. S. Huang and R. Y. Tsai, "Image sequence analysis: motion estimation," in *Image Sequence Analysis*, T. S. Huang, ed. (Springer-Verlag, Berlin, 1981).
9. C. Longuet-Higgins, "A computer algorithm for reconstructing a scene from two projections," Nature **293,** 133–135 (1981).
10. R. Y. Tsai and T. S. Huang, "Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces," Coordinated Science Laboratory Rep. R-921 (University of Illinois at Urbana-Champaign, Ill., October 1981).
11. R. Y. Tsai and T. S. Huang, "Uniqueness and estimation of 3-D motion parameters of rigid objects with curved surfaces," IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-6,** 13–17 (1984).
12. B. L. Yen and T. S. Huang, "Determining 3-D motion parameters of a rigid body: a vector-geometric approach," presented at the Association for Computing Machinery Workshop on Motion, Toronto, 1983.
13. X. Zhuang and R. M. Haralick, "Two-view motion analysis under a small perturbation," presented at the International Conference on Acoustics, Signal Processing, and Speech Understanding, Tampa, Fla., 1985.
14. C. Longuet-Higgins, "The reconstruction of a scene from two projections—configurations that defeat the 8-point algorithm," presented at the First Conference on Artificial Intelligence Applications, Denver, Colo., December 5–7, 1984.
15. S. Arun, S. D. Blostein, and T. S. Huang, "Least-squares fitting of 3-D point sets," Coordinated Science Laboratory/UI Memo (University of Illinois at Urbana-Champaign, Ill., 1986).