MASSACHUSETTS INSTITUTE OF TECHNOLOGY
THE ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 994 September 1987

Relative Orientation

Berthold K.P. Horn

Abstract: Before corresponding points in images taken with two cameras can be used to recover distances to objects in a scene, one has to determine the position and orientation of one camera relative to the other. This is the classic photogrammetric problem of *relative orientation*, central to the interpretation of binocular stereo information. Iterative methods for determining relative orientation were developed long ago; without them we would not have most of the topographic maps we do today. Relative orientation is also of importance in the recovery of motion and shape from an image sequence when successive frames are widely separated in time. Workers in motion vision are rediscovering some of the methods of photogrammetry.

Described here is a particularly simple iterative scheme for recovering relative orientation that, unlike existing methods, does not require a good initial guess for the baseline and the rotation. The data required is a set of pairs of corresponding rays from the two projection centers to points in the scene. It is well known that at least five pairs of rays are needed. Less appears to be known about the existence of multiple solutions and their interpretation. These issues are discussed here in detail. The unambiguous determination of all of the parameters of relative orientation is not possible when the observed points lie on a *critical surface*. These surfaces and their degenerate forms are analysed here as well.

Key Words: Relative Orientation, Photogrammetry, Binocular Stereo, Motion Vision, Coplanarity Condition, Representation of Rotation, Sampling the Space of Rotations.

Acknowledgements: This paper describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Army contract number DACA76-85-C-0010, and in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract number N00014-85-K-0124.

1. Introduction

The positions of corresponding points in two images can be used to determine the positions of points in the environment, provided that the position and orientation of one camera with respect to the other is known. Given the internal geometry of the camera, including its focal length and the location of the principal point, rays can be constructed by connecting the points in the images to their corresponding projection centers. These rays, when extended, intersect at the point in the scene that gave rise to the image points. This is how binocular stereo data is used to determine the positions of points in the environment after the correspondence problem has been solved.

It is also the method used in motion vision when feature points are tracked and the image displacements that occur in the time between two successive frames are relatively large (see for example [Ullman 1979] and [Tsai & Huang 1984]). The connection between these two problems has not attracted much attention before, nor has the relationship of motion vision to some aspects of photogrammetry (but see [Longuet-Higgins 1981]). It turns out, for example, that the well known motion field equations [Longuet-Higgins & Prazdny 1980, Bruss & Horn 1983] are just the *parallax equations* of photogrammetry [Hallert 1960, Moffit & Mikhail 1980] that occur in the incremental adjustment of relative orientation. Most papers on relative orientation only give the equation for y-parallax, corresponding to the equation for the y-component of the motion field (see for example the first equation in [Gill 1964], equation (1) in [Jochmann 1965], and equation (6) in [Oswal 1967]). Some papers actually give equations for both x- and y-parallax (see for example equation (9) in [Bender 1967]).

In both binocular stereo and large displacement motion vision analysis, it is necessary to first determine the *relative orientation* of one camera with respect to the other. The relative orientation can be found if a sufficiently large set of pairs of corresponding image points have been identified [Thompson 1959b, Thompson 1968, Ghosh 1972, Schwidefsky 1973, Slama et al. 1980, Moffit & Mikhail 1980, Wolf 1983, Horn 1986].

Let us use the terms *left* and *right* to distinguish the two cameras (in the case of the application to motion vision these will be the camera positions and orientations corresponding to the earlier and the later frames respectively).[1] The ray from the center of projection of the left camera to the center of projection of the right camera is called the *baseline* (see Fig. 1). A coordinate system can be erected at each projection center, with one axis along the

[1] In what follows we use the coordinate system of the right (or later) camera as the reference. One can simply interchange left and right if it happens to be more convenient to use the coordinate system of the left (or earlier) camera.

optical axis, that is, perpendicular to the image plane. The other two axes are in each case parallel to two convenient orthogonal directions in the image plane (such as the edges of the image, or lines connecting pairs of fiducial marks). The rotation of the left camera coordinate system with respect to the right is called the *orientation.*

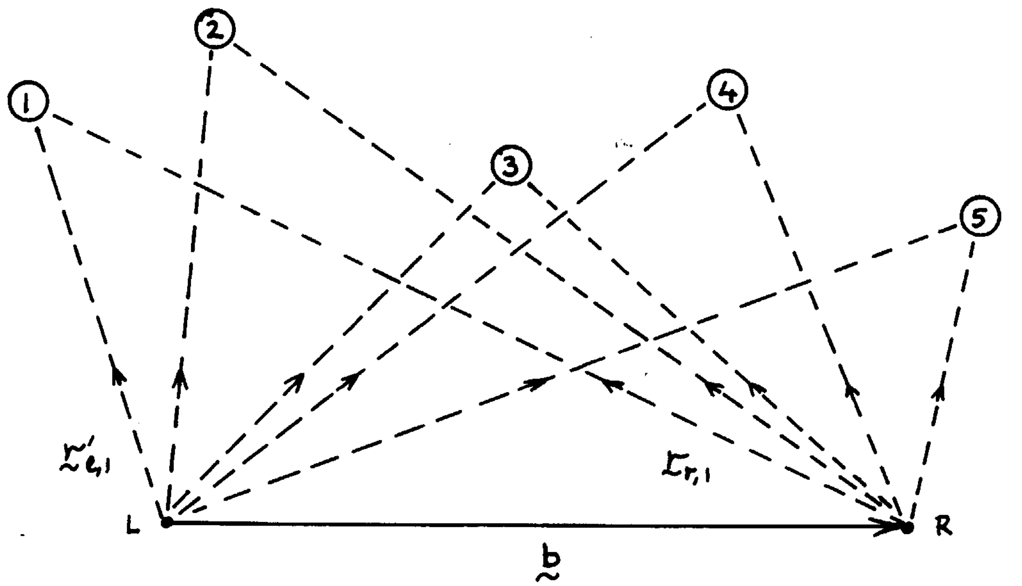

Figure 1. Points in the environment are viewed from two camera positions. The relative orientation is the direction of the baseline **b**, and the rotation R relating the left and right coordinate systems. The directions of rays to at least five scene points must be known in both camera coordinate systems.

Note that we cannot determine the length of the baseline without knowledge about the length of a line in the scene, since the ray directions are unchanged if we scale all of the distances in the scene and the baseline by the same positive scale-factor. This means that we should treat the baseline as a unit vector, and that there are really only five unknowns—three for the rotation and two for the direction of the baseline.[2]

2. Existing Solution Methods

Various empirical procedures have been devised for determining the relative orientation in an analog fashion. Most commonly used are stereoplotters, op-

[2]If we treat the baseline as a unit vector, its actual length becomes the unit of length for all other quantities.

tical devices that permit viewing of image pairs and superimposed synthetic features called floating marks. Differences in ray direction parallel to the baseline are called *horizontal disparities* (or x-parallaxes), while differences in ray direction orthogonal to the baseline are called *vertical disparities* (or y-parallaxes).[3] Horizontal disparities encode distances to points on the surface and are the quantities sought after in measurement of the underlying topography. There should be no vertical disparities when the device is adjusted to the correct relative orientation, since the rays from the left and right projection center must lie in a plane that contains the baseline if they are to intersect.

The methods used in practice to determine the correct relative orientation depend on successive adjustments to eliminate the vertical disparity at each of five or six carefully chosen points [Sailor 1960, Thompson 1964, Slama et al. 1980, Moffit & Mikhail 1980, Wolf 1974]. In each of the adjustments a single parameter of the relative orientation is varied in order to remove the vertical disparity at one of the points. Which adjustment is made to eliminate the vertical disparity at a particular point depends on the particular method is chosen. In each case, however, one of the adjustments, rather than being guided visually, is made by an amount that is calculated, using the measured values of earlier adjustments. The calculation is based on the assumptions that the surface being viewed can be approximated by a plane, that the baseline is roughly parallel to this plane, and that the optical axes of the two cameras are roughly perpendicular to this plane. The whole process is iterative in nature, since the reduction of vertical disparity at one point by means of an adjustment of a single parameter of the relative orientation disturbs the vertical disparity at the other points. Convergence is usually rapid if a good initial guess is available. It can be slow, however, when the assumptions on which the calculation is based are violated, such as in "accidented" or hilly terrain [Van Der Weele 1959–60]. These methods typically use Euler angles to represent three-dimensional rotations [Korn & Korn 1968] (traditionally denoted by the greek letters κ, ϕ, and ω). Euler angles have a number of short-comings for describing rotations that become particularly noticable when these angles become large.

There also exist related digital procedures that converge rapidly when a good initial guess of the relative orientation is available, as is usually the case when one is interpreting aerial photography [Slama et al. 1980]. Not all of these methods use Euler angles. Thompson [1959b], for example, uses twice the Gibb's vector [Korn & Korn 1968] to represent rotations. These proce-

[3]This naming convention stems from the observation that, roughly speaking, in the usual viewing arrangement, horizontal disparities correspond to left-right displacements in the image, whereas vertical disparities correspond to up-down displacements.

dures may fail to converge to the correct solution when the initial guess is far off the mark. In the application to motion vision, approximate translational and rotational components of the motion are often not known initially, so a procedure that depends on good initial guesses is not particularly useful.

Normally, the directions of the rays are obtained from points in images generated by projection onto a planar surface. In this case the directions are confined to the field of view as determined by the active area of the image plane and the distance to the center of projection. The field of view is always less than a hemi-sphere, since only points in front of the camera can be imaged.[4] The method described here applies, however, no matter how the directions to points in the scene are determined. There is no restriction on the possible directions of the rays. We do assume, however, that we can tell which of two opposite semi-infinite line-segments the point lies on. If a point lies on the correct line-segment we will say that it lies in *front* of the camera, otherwise it will be considered to be *behind* the camera.

The problem of relative orientation is generally considered solved, and so has received little attention in the photogrammetric literature in recent times [Van Der Weele 1959–60]. In the annual index of *Photogrammetric Engineering*, for example, there is only one reference to the subject in the last ten years [Ghilani 1983] and six in the decade before that. This is very little in comparison to the large number of papers on this subject in the fifties, as well as the sixties, including [Gill 1964], [Sailor 1965], [Jochmann 1965], [Ghosh 1966], [Forrest 1966] and [Oswal 1967].

In this paper we discuss the relationship of relative orientation to the problem of motion vision in the situation where the motion between the exposure of successive frames is relatively large. Also, a new iterative algorithm is described here, as well as a way of dealing with the situation when there is no initial guess available for the rotation or the direction of the baseline. The advantages of the unit quaternion notation for representing rotations are illustrated as well. Finally, we discuss critical surfaces, surface shapes that lead to difficulties in establishing a unique relative orientation.

3. Coplanarity Condition

If the ray from the left camera and the corresponding ray from the right camera are to intersect, they must to lie in a plane that also contains the baseline. Thus, if $\mathbf{b}$ is the vector respresenting the baseline, $\mathbf{r}_r$ the ray from the right projection center to the point in the scene and $\mathbf{r}_l$ the ray from the

[4]The field of view is larger than a hemi-sphere in some fish-eye lenses, where there is significant radial distortion.

left projection center to the point in the scene, then the triple product

$$[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r]$$

equals zero, where $\mathbf{r}_l' = \mathrm{Rot}(\mathbf{r}_l)$ is the left ray rotated into the right camera's coordinate system.[5] This is the *coplanarity condition* (see Fig. 2).

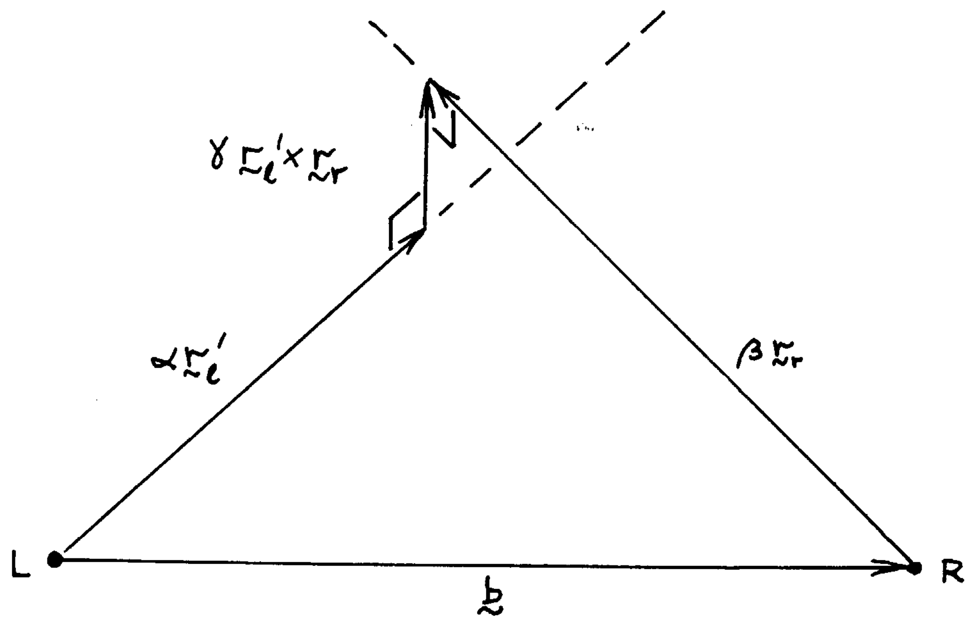

Figure 2. Two rays approach closest where they are intersected by a line perpendicular to both. If there is no measurement error, and the relative orientation has been recovered correctly, then the two rays actually intersect. In this case the two rays and the baseline lie in a common plane.

We obtain one such constraint from each pair of rays. There will be an infinite number of solutions for the baseline and the rotation when there are fewer than five pairs of rays, since there are five unknowns and each pair of rays yields only one constraint. Conversely, if there are more than five pairs of rays, the constraints are likely to be inconsistent as the result of small errors in the measurements. In this case, no exact solution of the set of constraint equations exists, and it makes sense instead to minimize the sum of squares of errors in the constraint equations. In practice, one should use more than five pairs of rays in order to reduce the influence of measurement errors [Jochmann 1965, Ghosh 1966]. We shall see later that the added information also allows one to eliminate spurious solutions.

[5]The baseline vector **b** is here also measured in the coordinate system of the right camera.

4. What is the Appropriate Error Term?

The triple product $[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]$ is zero when the left and right ray are coplanar with the baseline. It is not immediately apparent, however, that the triple product itself is necessarily the ideal measure of departure from best fit. It is worthwhile exploring the geometry of the two rays more carefully. Consider the points on the rays where they approach each other the closest (see Fig. 2). The line connecting these points will be perpendicular to both rays, and hence parallel to $(\mathbf{r}'_l \times \mathbf{r}_r)$. As a consequence, we can write

$$\alpha\,\mathbf{r}'_l + \gamma\,(\mathbf{r}'_l \times \mathbf{r}_r) = \mathbf{b} + \beta\,\mathbf{r}_r,$$

where α and β are proportional to the distances along the left and the right ray to the points where they approach most closely, while γ is proportional to the shortest distance between the rays. We can find γ by taking the dot-product of the equality above with $\mathbf{r}'_l \times \mathbf{r}_r$. We obtain

$$\gamma\,\|\mathbf{r}'_l \times \mathbf{r}_r\|^2 = [\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r].$$

Similarly, taking the dot-products with $\mathbf{r}_r \times (\mathbf{r}'_l \times \mathbf{r}_r)$ and $\mathbf{r}'_l \times (\mathbf{r}'_l \times \mathbf{r}_r)$, we obtain

$$\alpha\,\|\mathbf{r}'_l \times \mathbf{r}_r\|^2 = (\mathbf{b} \times \mathbf{r}_r) \cdot (\mathbf{r}'_l \times \mathbf{r}_r),$$

$$\beta\,\|\mathbf{r}'_l \times \mathbf{r}_r\|^2 = (\mathbf{b} \times \mathbf{r}'_l) \cdot (\mathbf{r}'_l \times \mathbf{r}_r).$$

The magnitudes of α and β are the distances along the rays to the point of closest approach when $\mathbf{r}_r$ and $\mathbf{r}'_l$ are unit vectors. It turns out, however, that we are more interested here in the signs than in the magnitudes of α and β.

Normally, the points where the two rays approach the closest will be in front of both cameras, that is, both α and β will be positive. If the estimated baseline or rotation is in error, however, then it is possible for one or both of the calculated parameters α and β to be negative. We will use this observation later to distinguish amongst different apparent solutions.

We have shown that the perpendicular distance between the left and the right ray is equal to ratio of the triple product $[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]$ to the magnitude squared of $(\mathbf{r}'_l \times \mathbf{r}_r)$. But the measurement errors are in the image, not in the scene. Thus a least-squares procedure should be based on an error in determining the direction of the rays, not on the distance of closest approach. To arrive at such a measure, we can look at the angle, θ say, between the projections of the left and right ray into a plane perpendicular to $\mathbf{b}$ (see Fig. 3). This angle will be zero when the vertical disparity is zero, that is, when the left and right rays are coplanar with the baseline.

The projections of $\mathbf{r}'_l$ and $\mathbf{r}_r$ into a plane perpendicular to $\mathbf{b}$ are given by

$$\overline{\mathbf{r}}'_l = \mathbf{r}'_l - (\mathbf{r}'_l \cdot \mathbf{b})\mathbf{b} = (\mathbf{b} \times \mathbf{r}'_l) \times \mathbf{b},$$

$$\overline{\mathbf{r}}_r = \mathbf{r}_r - (\mathbf{r}_r \cdot \mathbf{b})\mathbf{b} = (\mathbf{b} \times \mathbf{r}_r) \times \mathbf{b},$$

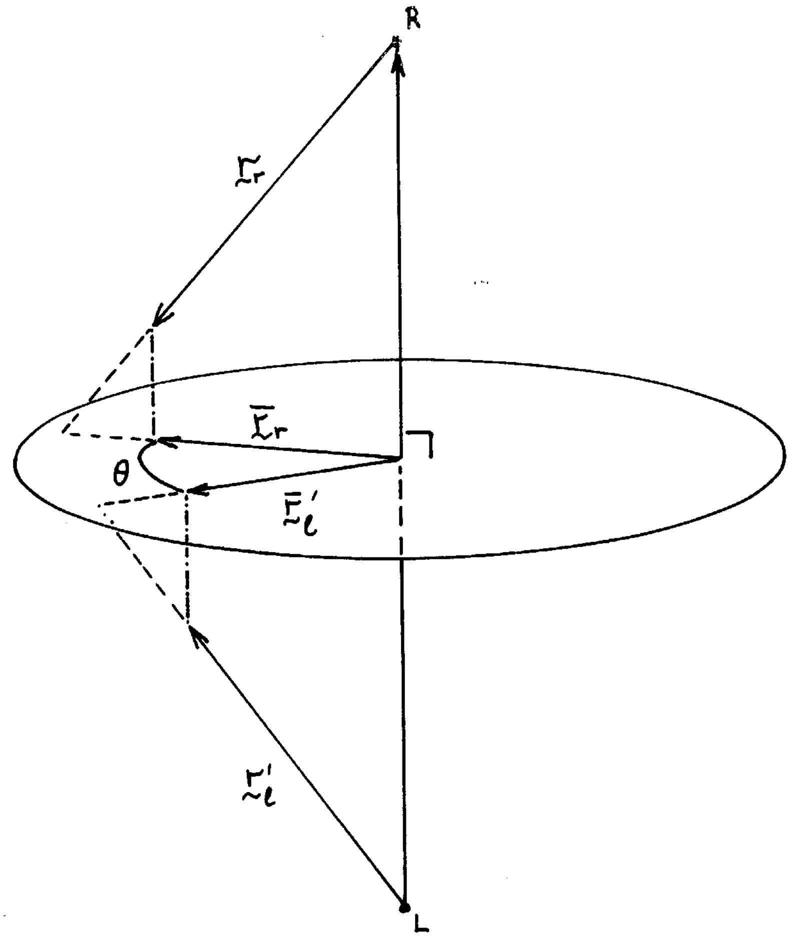

Figure 3. One measure of the departure from the coplanarity condition is the angle θ between the planes formed by the left ray and the baseline and the plane formed by the right ray and the baseline. The angle may be found by projection the two rays onto a plane perpendicular to the baseline.

where we have used the fact that $\mathbf{b}$ is a unit vector. It is easy to show that

$$\|\bar{\mathbf{r}}_l'\| = \|\mathbf{b} \times \mathbf{r}_l'\| \quad \text{and} \quad \|\bar{\mathbf{r}}_r\| = \|\mathbf{b} \times \mathbf{r}_r\|,$$

keeping in mind again that $\mathbf{b}$ is a unit vector, and that $\mathbf{b} \times \mathbf{r}_l'$ and $\mathbf{b} \times \mathbf{r}_r$ are perpendicular to $\mathbf{b}$. The cross-product of the two projected vectors $\bar{\mathbf{r}}_l'$ and $\bar{\mathbf{r}}_r$ will be parallel to the baseline and have magnitude proportional to the sine of the angle between the projected vectors, and so,

$$\sin\theta = \frac{[\mathbf{b}\ \bar{\mathbf{r}}_l'\ \bar{\mathbf{r}}_r]}{\|\bar{\mathbf{r}}_l'\|\ \|\bar{\mathbf{r}}_r\|} = \frac{[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r]}{\|\mathbf{b} \times \mathbf{r}_l'\|\ \|\mathbf{b} \times \mathbf{r}_r\|},$$

where we have used the fact that $[\mathbf{b}\ \bar{\mathbf{r}}_l'\ \bar{\mathbf{r}}_r] = [\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r]$, something that can easily be verified.

We could use the sine of the angle between the projected vectors directly as a measure of departure from best fit. This is not as good an idea as it may appear at first sight, because of what happens when one of the rays becomes nearly parallel to the baseline. In this case the angle will vary rapidly with small changes in the direction of the ray. Correspondingly, one of the terms in the denominator in the expression for the sine of the angle becomes small. It is better to normalize the expression by multiplying by the lengths of the projected vectors. Then we obtain the area of the parallelogram formed by the projections of the rays into a plane perpendicular to the baseline, namely,

$$\|\bar{\mathbf{r}}_l'\|\,\|\bar{\mathbf{r}}_r\|\sin\theta = [\mathbf{b}\ \bar{\mathbf{r}}_l'\ \bar{\mathbf{r}}_r] = [\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r].$$

This discussion confirms that the triple product itself is a good measure of the departure from best fit. This is convenient, since it makes the least squares analysis reasonably straightforward. If one desires to use a different error measure, one can weight the terms in the sums to follow accordingly.

5. Least Squares Solution for the Baseline

If the rotation is known, it is easy to find the best fit baseline, as we show next. This is useful, despite the fact that we do not usually know the rotation. The reason is that the ability to find the best baseline, given a rotation, reduces the dimensionality of the search space from five to three.

Let $\{\mathbf{r}_{l,i}\}$ and $\{\mathbf{r}_{r,i}\}$, for $i = 1 \ldots n$, be corresponding sets of left and right rays. We wish to minimize

$$E = \sum_{i=1}^{n} [\mathbf{b}\ \mathbf{r}_{l,i}'\ \mathbf{r}_{r,i}]^2 = \sum_{i=1}^{n} \left(\mathbf{b} \cdot (\mathbf{r}_{l,i}' \times \mathbf{r}_{r,i})\right)^2,$$

subject to the condition $\|\mathbf{b}\|^2 = 1$, where $\mathbf{r}_{l,i}'$ is the rotated left ray $\mathbf{r}_{l,i}$, as before. If we let $\mathbf{c}_i = \mathbf{r}_{l,i}' \times \mathbf{r}_{r,i}$, we can rewrite the sum in the simpler form

$$E = \sum_{i=1}^{n} (\mathbf{b} \cdot \mathbf{c}_i)^2 = \mathbf{b}^T \left(\sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^T \right) \mathbf{b},$$

where we have used the equivalence $\mathbf{b} \cdot \mathbf{c}_i = \mathbf{b}^T \mathbf{c}_i$, which depends on the interpretation of column vectors as 3×1 matrices. The term $\mathbf{c}_i \mathbf{c}_i^T$ is a dyadic product, a 3×3 matrix obtained by multiplying a 3×1 matrix by a 1×3 matrix.

The error sum is a quadratic form involving the real symmetric matrix

$$C = \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^T.$$

The minimum of such a quadratic form is the smallest eigenvalue of the matrix C, attained when $\mathbf{b}$ is the corresponding unit eigenvector (see, for example, the discussion of Rayleigh's quotient in [Korn & Korn 1968]). This can be verified by introducing a Lagrangian multiplier λ and minimizing

$$E' = \mathbf{b}^T C\, \mathbf{b} + \lambda(1 - \mathbf{b}^T\mathbf{b}),$$

subject to the condition $\mathbf{b}^T\mathbf{b} = 1$. Differentiating with respect to $\mathbf{b}$ and setting the result equal to zero yields:

$$C\,\mathbf{b} = \lambda\mathbf{b}.$$

The error corresponding to a particular solution of this equation is found by premultiplying by $\mathbf{b}^T$:

$$E = \mathbf{b}^T C\, \mathbf{b} = \lambda\mathbf{b}^T\mathbf{b} = \lambda.$$

The three eigenvalues of the real symmetric matrix C are non-negative, and can be found in closed form by solving a cubic equation, while each of the corresponding eigenvectors has components that are the solution of three homogeneous equations in three unknowns [Korn & Korn 1968]. If the data are relatively free of measurement error, then the smallest eigenvalue will be much smaller than the other two, and a reasonable approximation to the sought-after result can be obtained by solving for the eigenvector using the assumption that the smallest eigenvalue is actually zero. This way one need not even solve the cubic equation (see also [Horn & Weldon 1988]).

If $\mathbf{b}$ is a unit eigenvector, so is $-\mathbf{b}$. Changing the sense of the baseline does not change the magnitude of the error term $[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]$. It does, however, change the signs of α, β and γ. One can decide which sense of the baseline direction is appropriate by determining the signs of α_i and β_i for $i = 1 \ldots n$. Ideally, they should all be positive. The solution for the optimal baseline is not unique unless there are at least two pairs of corresponding rays. The reason is that the eigenvector we are looking for is not uniquely determined if more than one of the eigenvalues is zero, and the matrix has rank less than two if it is the sum of fewer than two dyadic products of independent vectors. This is not a significant restriction, however, since we need at least five pairs of rays to solve for the rotation anyway.

6. Iterative Improvement of Relative Orientation.

If one ignores the orthonormality of the rotation matrix, a set of nine homogeneous linear equations can be obtained by a transformation of the coplanarity conditions that was first described in [Thompson 1959b]. These equations can be solved when eight pairs of corresponding ray directions are known [Longuet-Higgins 1981]. This is not a least-squares method that can make use of redundant measurements, nor can it be applied when fewer than eight points are given. The method is also strongly affected by measurement errors and fails for certain configurations of points [Longuet-Higgins 1984].

No true closed-form solution of the least-squares problem has been found for the general case, where both baseline and rotation are unknown. However, it is possible to determine how the overall error is affected by small changes in the baseline and small changes in the rotation. This allows one to make iterative adjustments to the baseline and the rotation that reduce the sum of squares of errors.

We can represent a small change in the baseline by an infinitesimal quantity $\delta\mathbf{b}$. If this change is to leave the length of the baseline unaltered, then

$$\|\mathbf{b} + \delta\mathbf{b}\|^2 = \|\mathbf{b}\|^2,$$

or

$$\|\mathbf{b}\|^2 + 2\mathbf{b} \cdot \delta\mathbf{b} + \|\delta\mathbf{b}\|^2 = \|\mathbf{b}\|^2.$$

If we ignore quantities of second-order, we obtain

$$\delta\mathbf{b} \cdot \mathbf{b} = 0,$$

that is, $\delta\mathbf{b}$ must be perpendicular to $\mathbf{b}$.

A small change in the rotation can be represented by an infinitesimal rotation vector $\delta\boldsymbol{\omega}$. The direction of this vector is parallel to the axis of rotation, while its magnitude is the angle of rotation. This incremental rotation takes the rotated left ray, $\mathbf{r}'_l$, into

$$\mathbf{r}''_l = \mathbf{r}'_l + (\delta\boldsymbol{\omega} \times \mathbf{r}'_l).$$

This follows from Rodrigues' formula for the rotation of a vector $\mathbf{r}$:

$$\cos\theta\ \mathbf{r} + \sin\theta\,(\boldsymbol{\omega} \times \mathbf{r}) + (1 - \cos\theta)(\boldsymbol{\omega} \cdot \mathbf{r})\,\boldsymbol{\omega},$$

if we let $\theta = \|\delta\boldsymbol{\omega}\|$, $\boldsymbol{\omega} = \delta\boldsymbol{\omega}/\|\delta\boldsymbol{\omega}\|$, and note that $\delta\boldsymbol{\omega}$ is an infinitesimal quantity. Finally then, we see that $[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]$ becomes

$$[(\mathbf{b} + \delta\mathbf{b})\ \ (\mathbf{r}'_l + \delta\boldsymbol{\omega} \times \mathbf{r}'_l)\ \ \mathbf{r}_r],$$

or,

$$[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]\ +\ [\delta\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]\ +\ [\mathbf{b}\ (\delta\boldsymbol{\omega} \times \mathbf{r}'_l)\ \mathbf{r}_r],$$

if we ignore the term $[\delta\mathbf{b}\ (\delta\boldsymbol{\omega} \times \mathbf{r}'_l)\ \mathbf{r}_r]$, because it contains the product of two infinitesimal quantities. We can expand two of the triple products in the expression above and obtain

$$[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r] + (\mathbf{r}'_l \times \mathbf{r}_r) \cdot \delta\mathbf{b} + (\mathbf{r}_r \times \mathbf{b}) \cdot (\delta\boldsymbol{\omega} \times \mathbf{r}'_l),$$

or

$$t + \mathbf{c} \cdot \delta\mathbf{b} + \mathbf{d} \cdot \delta\boldsymbol{\omega},$$

for short, where

$$t = [\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r], \quad \mathbf{c} = \mathbf{r}'_l \times \mathbf{r}_r, \quad \text{and} \quad \mathbf{d} = \mathbf{r}'_l \times (\mathbf{r}_r \times \mathbf{b}).$$

Now, we are trying to minimize

$$E = \sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})^2,$$

subject to the condition $\mathbf{b} \cdot \delta\mathbf{b} = 0$. This constrained minimization problem can be transformed into an equivalent unconstrained form by introduction of a Lagrangian multiplier. Instead of minimizing E itself, we then have to minimize:

$$E' = E + \lambda(\mathbf{b} \cdot \delta\mathbf{b}).$$

Differentiating E' with respect to $\delta\mathbf{b}$, and setting the result equal to zero yields,

$$\sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})\,\mathbf{c}_i + \lambda\mathbf{b} = \mathbf{0}.$$

Before we can proceed, we need to eliminate the unknown Lagrangian multiplier λ. The dot-product of the expression with $\mathbf{b}$ leads to

$$\sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})\,\mathbf{c}_i \cdot \mathbf{b} + \lambda(\mathbf{b} \cdot \mathbf{b}) = 0,$$

which, since $\mathbf{b} \cdot \mathbf{b} = 1$, gives us a value for λ that we can use to compute the term

$$\lambda\mathbf{b} = -\mathbf{b}\sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})\,\mathbf{c}_i \cdot \mathbf{b},$$

or

$$\lambda\mathbf{b} = -(\mathbf{b}\mathbf{b}^T)\sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})\,\mathbf{c}_i.$$

Finally, substituting for $\lambda\mathbf{b}$ in the equation above we obtain

$$B\sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})\,\mathbf{c}_i = \mathbf{0},$$

where

$$B = (I - \mathbf{b}\mathbf{b}^T)$$

is the projection operator that removes components of vectors parallel to the baseline, and I is just the 3×3 identity matrix. We can conclude that the equation above relates quantities in a plane perpendicular to the baseline.

Finally, if we differentiate E' with respect to $\delta\boldsymbol{\omega}$ and set this result also equal to zero, we obtain

$$\sum_{i=1}^{n} (t_i + \mathbf{c}_i \cdot \delta\mathbf{b} + \mathbf{d}_i \cdot \delta\boldsymbol{\omega})\,\mathbf{d}_i = \mathbf{0}.$$

Together, the two vector equations constitute six linear scalar equations in the six unknown components of $\delta\mathbf{b}$ and $\delta\boldsymbol{\omega}$. We can rewrite them in the more compact form:

$$\begin{aligned} BC\ \delta\mathbf{b} + BF\ \delta\boldsymbol{\omega} &= -B\,\overline{\mathbf{c}} \\ F^T\ \delta\mathbf{b}\ +\ D\ \delta\boldsymbol{\omega} &= -\overline{\mathbf{d}} \end{aligned}$$

or

$$\begin{pmatrix} BC & BF \\ F^T & D \end{pmatrix} \begin{pmatrix} \delta\mathbf{b} \\ \delta\boldsymbol{\omega} \end{pmatrix} = -\begin{pmatrix} B\,\bar{\mathbf{c}} \\ \bar{\mathbf{d}} \end{pmatrix}$$

where

$$C = \sum_{i=1}^{n} \mathbf{c}_i \mathbf{c}_i^T, \quad F = \sum_{i=1}^{n} \mathbf{c}_i \mathbf{d}_i^T, \quad \text{and} \quad D = \sum_{i=1}^{n} \mathbf{d}_i \mathbf{d}_i^T,$$

while

$$\bar{\mathbf{c}} = \sum_{i=1}^{n} t_i \mathbf{c}_i \quad \text{and} \quad \bar{\mathbf{d}} = \sum_{i=1}^{n} t_i \mathbf{d}_i.$$

The matrix B is singular, as can be seen by noting that $\mathbf{b}$ is an eigenvector with zero eigenvalue. Thus the first three equations above are not independent. One of them will have to be removed. Fortunately, we also still have to incorporate the condition that $\delta\mathbf{b}$ be perpendicular to $\mathbf{b}$. We can do this by replacing one of the first three equations with the linear equation $\mathbf{b} \cdot \delta\mathbf{b} = 0$. For the best numerical accuracy one should eliminate the equation with the smallest coefficients.

The above gives us a way of finding small changes in the baseline and rotation that reduce the overall error sum. This method can be applied iteratively to locate a minimum. Numerical experiments confirm that it converges rapidly when a good initial guess is available. Incremental adjustments cannot be computed if the six-by-six coefficient matrix becomes singular. This will happen when there are fewer than five pairs of rays, and for certain rare configurations of points in the scene (see the discussion of *critical surfaces* later on).

7. Adjusting the Baseline and the Rotation

The iterative adjustment of the baseline is straightforward:

$$\mathbf{b}^{n+1} = \mathbf{b}^n + \delta\mathbf{b}^n,$$

where $\mathbf{b}^n$ is the baseline estimate at the beginning of the n-th iteration, while $\delta\mathbf{b}^n$ is the adjustment computed during the n-th iteration, as discussed in the previous section. If $\delta\mathbf{b}^n$ is not infinitesimal, the result will not be a unit vector. We can, and should, normalize the result by dividing by its magnitude.

7.1. Adjustment of Rotation using Unit Quaternions

Adjusting the rotation is a little harder. Rotations are conveniently represented by unit quaternions [Stuelpnagle 1964, Salamin 1979, Taylor 1982, Horn 1986, Horn 1987a]. The groundwork for the application of the unit quaternion notation in photogrammetry was laid by Thompson [1959a] and Schut [1958–59]. A positive rotation about the axis $\boldsymbol{\omega}$ through an angle θ is represented by the unit quaternion

$$\mathring{q} = \cos(\theta/2) + \sin(\theta/2)\,\boldsymbol{\omega},$$

where $\boldsymbol{\omega}$ is assumed to be a unit vector. Composition of rotations corresponds to multiplication of the corresponding unit quaternions. The rotated version of a vector $\mathbf{r}$ is computed using

$$\mathring{r}' = \mathring{q}\,\mathring{r}\,\mathring{q}^*,$$

where $\mathring{q}^*$ is the conjugate of the quaternion $\mathring{q}$, that is, the quaternion obtained by changing the sign of the vector part. Here, $\mathring{r}$ is a purely imaginary quaternion with vector part $\mathbf{r}$, while $\mathring{r}'$ is a purely imaginary quaternion with vector part $\mathbf{r}'$.

The infinitesimal rotation $\delta\boldsymbol{\omega}$ corresponds to the quaternion

$$\delta\mathring{\omega} = 1 + \delta\boldsymbol{\omega}.$$

We can adjust the rotation $\mathring{q}$ by premultiplying with $\delta\mathring{\omega}$, that is,

$$\mathring{q}^{n+1} = \delta\mathring{\omega}^n\,\mathring{q}^n.$$

If $\delta\boldsymbol{\omega}^n$ is not infinitesimal, $\delta\mathring{\omega}^n$ will not be a unit quaternion, and so the result of the adjustment will not be a unit quaternion either. This undesirable state of affairs can be avoided by using either of the two unit quaternions

$$\delta\mathring{\omega} = \sqrt{1 - \|\delta\boldsymbol{\omega}\|^2} + \delta\boldsymbol{\omega},$$

or

$$\delta\mathring{\omega} = (1 + \delta\boldsymbol{\omega})/\sqrt{1 + \|\delta\boldsymbol{\omega}\|^2}.$$

Alternatively, one can simply normalize the product by dividing by its magnitude.

7.2. Adjustment of Rotation using Orthonormal Matrices

The adjustment of rotation is a little trickier if orthonormal matrices are used to represent rotations. We can write the relationship

$$\mathbf{r}' = \mathbf{r} + (\delta\boldsymbol{\omega} \times \mathbf{r}),$$

in the form

$$\mathbf{r}' = \mathbf{r} + W\,\mathbf{r},$$

where the skew-symmetric matrix W is defined by

$$W = \begin{pmatrix} 0 & -\delta\omega_z & \delta\omega_y \\ \delta\omega_z & 0 & -\delta\omega_x \\ -\delta\omega_y & \delta\omega_x & 0 \end{pmatrix},$$

in terms of the components of rotation vector $\delta\omega = (\delta\omega_x, \delta\omega_y, \delta\omega_z)^T$. Consequently we may write $\mathbf{r}' = Q\,\mathbf{r}$, where $Q = I + W$, or

$$Q = \begin{pmatrix} 1 & -\delta\omega_z & \delta\omega_y \\ \delta\omega_z & 1 & -\delta\omega_x \\ -\delta\omega_y & \delta\omega_x & 1 \end{pmatrix},$$

One could then attempt to adjust the rotation by multiplication of the matrices Q and R as folows:

$$R^{n+1} = Q^n R^n.$$

The problem is that Q is not orthonormal unless $\delta\omega$ is infinitesimal. In practice this means that the rotation matrix will depart more and more from orthonormality as more and more iterative adjustments are made. It is possible to re-normalize this matrix by finding the nearest orthonormal matrix, but this is complicated, involving the determination of the square-root of a symmetric matrix [Horn, Hilden & Negahdaripour 1988]. This is one place where the unit quaternion representation has a distinct advantage: it is trivial to find the nearest unit quaternion to a quaternion that does not have unit magnitude.

To avoid this problem, we should really start with an orthonormal matrix to represent the incremental rotation. We can use either of the unit quaternions

$$\delta\mathring{\omega} = \sqrt{1 - \|\delta\omega\|^2} + \delta\omega,$$

or

$$\delta\mathring{\omega} = (1 + \delta\omega)/\sqrt{1 + \|\delta\omega\|^2},$$

to construct the corresponding orthonormal matrix

$$Q = \begin{pmatrix} q_0^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_0 q_z) & 2(q_x q_z + q_0 q_y) \\ 2(q_y q_x + q_0 q_z) & q_0^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_0 q_x) \\ 2(q_z q_x - q_0 q_y) & 2(q_z q_y + q_0 q_x) & q_0^2 - q_x^2 - q_y^2 + q_z^2 \end{pmatrix},$$

where q_0 is the scalar part of the quaternion $\delta\mathring{\omega}$, while q_x, q_y, q_z are the components of the vector part. Then the adjustment of rotation is accomplished using

$$R^{n+1} = Q^n R^n.$$

Note, however, that the resulting matrices will still tend to depart somewhat from orthonormality due to numerical inaccuracies. This may be a problem if many iterations are required.

8. Inherent Ambiguities

The iterative adjustment described above may arrive at a number of apparently different solutions. Some of these are just different representations of the same solution, while others are related to the correct solution by a simple transformation. First of all, note that $-\mathring{q}$ represents the same rotation as $\mathring{q}$, since

$$(-\mathring{q})\,\mathring{r}\,(-\mathring{q}^*) = \mathring{q}\,\mathring{r}\,\mathring{q}^*.$$

That is, antipodal points on the unit sphere in four dimensions represent the same rotation. We can prevent any confusion by ensuring that the first non-zero component of the resulting unit quaternion is positive, or that the largest component is positive.

Next, note that the triple product, $[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r]$, changes sign, but not magnitude, when we replace $\mathbf{b}$ with $-\mathbf{b}$. Thus the two possible senses of the baseline yield the same sum of squares of errors. However, changing the sign of $\mathbf{b}$ does change the signs of both α and β. All scene points imaged are in front of the camera, so the distances should all be positive. In the presence of noise, it is possible that some of the distances turn out to be negative, but with reasonable data almost all of them should be positive. This allows us to easily pick the correct sense for the baseline.

Not so obvious is another possibility: Suppose we turn all of the left measurements through π radians about the baseline, in addition to the rotation already determined. That is, replace $\mathbf{r}_l'$ with

$$\mathbf{r}_l'' = 2(\mathbf{b}\cdot\mathbf{r}_l')\mathbf{b} - \mathbf{r}_l'.$$

This follows from Rodrigues' formula for the rotation of a vector $\mathbf{r}$:

$$\cos\theta\ \mathbf{r} + \sin\theta\,(\boldsymbol{\omega}\times\mathbf{r}) + (1-\cos\theta)(\boldsymbol{\omega}\cdot\mathbf{r})\,\boldsymbol{\omega},$$

with $\theta = \pi$ and $\boldsymbol{\omega} = \mathbf{b}$. Then the triple product $[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r]$ turns into

$$2(\mathbf{b}\cdot\mathbf{r}_l')[\mathbf{b}\ \mathbf{b}\ \mathbf{r}_r] - [\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r] = -[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r].$$

This, once again, reverses the sign of the error term, but not its magnitude. Thus the sum of squares of errors is unaltered. The signs of α and β are affected, however, although this time not in as simple a way as when the sense of the baseline was reversed.

If $[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r] = 0$, we find that exactly one of α and β changes sign. This can be shown as follows: The triple product will be zero when the left and right rays are coplanar with the baseline. In this case we have $\gamma = 0$, and so

$$\alpha\,\mathbf{r}_l' = \mathbf{b} + \beta\,\mathbf{r}_r,$$

Taking the cross-product with $\mathbf{b}$ we obtain

$$\alpha\,(\mathbf{r}_l'\times\mathbf{b}) = \beta\,(\mathbf{r}_r\times\mathbf{b}),$$

If we now replace $\mathbf{r}_l'$ by $\mathbf{r}_l'' = 2(\mathbf{b}\cdot\mathbf{r}_l')\mathbf{b} - \mathbf{r}_l'$, we have for the new distances α' and β' along the rays:

$$-\alpha'\,(\mathbf{r}_l'\times\mathbf{b}) = \beta'\,(\mathbf{r}_r\times\mathbf{b}),$$

We conclude that the product $\alpha'\beta'$ has sign opposite to that of the product $\alpha\beta$. So if α and β are both positive, one of α' or β' must be negative.

In the presence of measurement error the triple product will not be exactly equal to zero. If the rays are nearly coplanar with the baseline, however, we find that one of α and β almost always changes sign. With very poor data, it is possible that both change sign. (Even with totally random ray directions, however, this only happens 27.3% of the time, as determined by Monte Carlo methods). In any case, we can reject a solution in which roughly half the distances are negative. Moreover, we can find the correct solution directly by introducing an additional rotation of π radians about the baseline.

9. Remaining Ambiguity

If we take care of the three apparent two-way ambiguities discussed in the previous section, we find that in practice a unique solution is found, provided that a sufficiently large number of ray pairs are available. That is, the method converges to the unique global minimum from every possibly starting point in parameter space.

Local minima in the sum of squares of errors appear when only a few more than the minimum of five ray pairs are available (as is common in practice). This means that one has to repeat the iteration with different starting values for the rotation in order to locate the global minimum. A starting value for the baseline can be found in each case using the closed-form method described in section 5. To search the parameter space effectively, one needs a way of efficiently sampling the space of rotations. The space of rotations is isomorphic to the unit sphere in four dimensions, with antipodal points identified. The rotation groups of the regular polyhedra provide convenient means of uniformly sampling the space of rotations. The group of rotations of the tetrahedron has 12 elements, that of the hexahedron and the octahedron has 24, and that of the icosahedron and the dodecahedron has 60 (representations of these groups are given in Appendix A for convenience). One can use these as starting values for the rotation. Alternatively, one can just generate a number of randomly placed points on the unit sphere in four dimensions as starting values for the rotation.

When there are only five pairs of rays, the situation is different again. In this case, we have five non-linear equations in five unknowns and so in general expect to find a finite number of exact solutions. That is, it is possible to find baselines and rotations that satisfy the coplanarity conditions exactly and reduce the sum of squares of errors to zero. If we ignore the three types of ambiguities discussed above (incuding the rotation by π radians about the baseline), then there are generally four distinct sets of baselines and rotations that yield an exact solution. Typically only one of these yields positive signs for all of the distances. These are empirical observations; I have not been able to prove that there are generally four solutions that satisfy the coplanarity conditions.

10. Summary of the Algorithm

Consider first the case where we have an initial guess for the rotation. We start by finding the best-fit baseline direction using the closed-form method described in section 5. We determine the correct sense of the baseline by choosing the one that makes most of the signs of the distances positive. Then we proceed as follows:

- For each pair of corresponding image points, we compute $\mathbf{r}'_{l,i}$, the left ray direction $\mathbf{r}_{l,i}$ rotated into the right camera coordinate system, using the present guess for the rotation.
- We then compute the cross-product $\mathbf{c}_i = \mathbf{r}'_{l,i} \times \mathbf{r}_{r,i}$, the double cross-product $\mathbf{d}_i = \mathbf{r}'_{l,i} \times (\mathbf{r}_{r,i} \times \mathbf{b})$ and the triple-product $t_i = [\mathbf{b}\ \mathbf{r}'_{l,i}\ \mathbf{r}_{r,i}]$.
- We accumulate the dyadic products $\mathbf{c}_i\mathbf{c}_i^T$, $\mathbf{c}_i\mathbf{d}_i^T$ and $\mathbf{d}_i\mathbf{d}_i^T$, as well as the vectors $t_i\mathbf{c}_i$ and $t_i\mathbf{d}_i$. The totals of these quantities over all image point pairs give us the matrices C, F, D and the vectors $\overline{\mathbf{c}}$ and $\overline{\mathbf{d}}$.
- We can now solve for the increment in the baseline $\delta\mathbf{b}$ and the increment in the rotation $\delta\boldsymbol{\omega}$ using the method derived in section 6.
- We adjust the baseline and the rotation using the methods discussed in section 7, and recompute the sum of the squares of the error terms.

The new orientation parameters are then used in the next iteration of the above sequence of steps. As is the case with most iterative procedures, it is hard to decide when to stop. Typically, the total error becomes small after a few iterations and no longer decreases at each step, because of limited accuracy in the arithmetic operations. One can stop the iteration the first time the error increases. Alternatively, one can stop after either a fixed number of iterations or after the error becomes less then some predetermined threshold.

When the decision has been made to stop the iteration, a check of the signs of the distances along the rays is in order. If most of them are negative, the baseline direction should be reversed. If neither sense of the baseline direction yields mostly positive distances, one needs to consider the possibility of a rotation through π radians about the baseline $\mathbf{b}$. If this also yields mixed signs, one is dealing with a local extremum of the error function; something that will only happen if the initial guess is in fact not adequate.

If an initial guess is not available, one proceeds as follows:

- For each rotation in the chosen group of rotations, perform the above iteration to obtain a candidate baseline and rotation.
- Choose the solution that yields the smallest total error.

When there are many pairs of rays, the iterative algorithm will converge to the global minimum error solution from any initial guess for the rotation. There is no need to sample the space of rotations in this case. Also, instead of sampling the space of rotations in a systematic way using a finite group of rotations, one can generate points randomly distributed on the surface of the unit sphere in four-dimensional space. This provides a simpler means of generating initial guesses, although more initial guesses may have to be tried than when a systematic procedure is used to sample the space of rotations.

The method given minimizes the sum of the squares of the triple products

$$[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r].$$

If desired, one can modify it to use some multiple of the triple product as an error term by weighting the contribution to the overall sum. This can lead to a considerably more complex optimization problem if the weights depend on the unknown baseline and the unknown rotation. This happens, for example, if we try to minimize the sum of squares of the sines of the angles corresponding to vertical disparity:

$$\sin\theta = \frac{[\mathbf{b}\ \mathbf{r}_l'\ \mathbf{r}_r]}{\|\mathbf{b}\times\mathbf{r}_l'\|\,\|\mathbf{b}\times\mathbf{r}_r\|}.$$

If we assume that the weighting factors vary slowly during the iterative process, however, we can to use the current estimates of the baseline and rotation in computing the weighting factors, and not take into account the small variations in the error sum due to changes in the weighting factors. That is, when taking derivatives, the weighting factors are considered constant. This is a good approximation when the parameters vary slowly, as they will when one is close to an extremum.

11. Critical Surfaces

In certain rare cases, relative orientation cannot be recovered fully, even when there are five or more pairs of rays. Normally, each error term varies linearly with distance in parameter space from the location of an extremum, and so the sum of squares of errors varies quadratically. There are situations, however, where the error terms to not vary linearly with distance, but quadratically or higher order, in certain special directions in parameter space. In this case, the sum of squares of errors does not vary quadratically with distance from the extremum, but as a function of the fourth or higher power of this distance.

This makes it very difficult to accurately locate the extremum. In an extreme situation, the total error may not vary at all along some curve in parameter space. In this case, the total error is unaffected by a small change in the rotation, as long as this change is accompanied by a corresponding small change in the baseline. There is no localized extremum and consequently the solution for relative orientation is not unique. It turns out that this problem arises only when the observed scene points lie on certain surfaces called *Gefährliche Flächen* or *critical surfaces* [Brandenberger 1947, Hofmann 1949, Zeller 1952, Schwidefsky 1973]. We show next that only points on certain hyperboloids of one sheet and their degenerate forms can lead to this kind of problem (see also [Horn 1987b]).

We could try to find a direction of movement in parameter space $(\delta\mathbf{b}, \delta\boldsymbol{\omega})$ that leaves the total error unaffected when given a particular surface. Instead, we will take the critical direction of motion in the parameter space as given, and try to find a surface for which the total error is unchanged.

Let $\mathbf{R}$ be a point on the surface, measured in the right camera coordinate system. Then

$$\beta\,\mathbf{r}_r = \mathbf{R} \quad \text{and} \quad \alpha\,\mathbf{r}'_l = \mathbf{b} + \mathbf{R},$$

for some positive α and β. In the absence of measurement errors,

$$[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r] = \frac{1}{\alpha\beta}[\mathbf{b}\ (\mathbf{b}+\mathbf{R})\ \mathbf{R}] = 0.$$

We noted earlier that when we change the baseline and the rotation slightly, the triple product $[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r]$ becomes

$$[(\mathbf{b}+\delta\mathbf{b})\ (\mathbf{r}'_l + \delta\boldsymbol{\omega}\times\mathbf{r}'_l)\ \mathbf{r}_r],$$

or, if we ignore higher-order terms,

$$[\mathbf{b}\ \mathbf{r}'_l\ \mathbf{r}_r] + (\mathbf{r}'_l \times \mathbf{r}_r)\cdot\delta\mathbf{b} + (\mathbf{r}_r\times\mathbf{b})\cdot(\delta\boldsymbol{\omega}\times\mathbf{r}'_l).$$

The problem we are focusing on here arises when this error term is unchanged for small movement in some direction in the parameter space. That is when

$$(\mathbf{r}'_l \times \mathbf{r}_r)\cdot\delta\mathbf{b} + (\mathbf{r}_r\times\mathbf{b})\cdot(\delta\boldsymbol{\omega}\times\mathbf{r}'_l) = 0,$$

for some $\delta\mathbf{b}$ and $\delta\boldsymbol{\omega}$. Introducing the coordinates of the imaged points we obtain:

$$\frac{1}{\alpha\beta}\Big(\big((\mathbf{b}+\mathbf{R})\times\mathbf{R}\big)\cdot\delta\mathbf{b} + (\mathbf{R}\times\mathbf{b})\cdot\big(\delta\boldsymbol{\omega}\times(\mathbf{b}+\mathbf{R})\big)\Big) = 0,$$

or

$$(\mathbf{R}\times\mathbf{b})\cdot(\delta\boldsymbol{\omega}\times\mathbf{R}) + (\mathbf{R}\times\mathbf{b})\cdot(\delta\boldsymbol{\omega}\times\mathbf{b}) + [\mathbf{b}\ \mathbf{R}\ \delta\mathbf{b}] = 0.$$

If we expand the first of the dot-products of the cross-products, we can write this equation in the form

$$(\mathbf{R}\cdot\mathbf{b})(\delta\boldsymbol{\omega}\cdot\mathbf{R}) - (\mathbf{b}\cdot\delta\boldsymbol{\omega})(\mathbf{R}\cdot\mathbf{R}) + \mathbf{L}\cdot\mathbf{R} = 0,$$

where

$$\mathbf{L} = \boldsymbol{\ell} \times \mathbf{b}, \quad \text{while} \quad \boldsymbol{\ell} = \mathbf{b} \times \delta\boldsymbol{\omega} + \delta\mathbf{b}.$$

The expression on the left-hand side contains a part that is quadratic in $\mathbf{R}$ and a part that is linear. The expression is clearly quadratic in X, Y, and Z, the components of the vector $\mathbf{R} = (X, Y, Z)^T$. Thus a surface leading to the kind of problem described above must be a quadric surface [Korn & Korn 1968].

Note that there is no constant term in the equation of the surface, so $\mathbf{R} = \mathbf{0}$ satisfies the equation. This means that the surface passes through the right projection center. It is easy to verify that $\mathbf{R} = -\mathbf{b}$ satisfies the equation also, which means that the surface passes through the left projection center as well. In fact, the whole baseline (and its extensions), $\mathbf{R} = k\mathbf{b}$, lies in the surface. This means that we must be dealing with a ruled quadric surface. It can consequently not be an ellipsoid or hyperboloid of two sheets, or one of their degenerate forms. The surface must be a hyperboloid of one sheet, or one of its degenerate forms. Additional information about the properties of these surfaces is given in Appendix B, while the degenerate forms are explored in Appendix C.

It should be apparent that this kind of ambiguity is quite rare. This is nevertheless an issue of practical importance, however, since the accuracy of the solution is reduced if the points lie near some critical surface. A textbook case of this occurs in aerial photography of a roughly U-shaped valley taken along a flight line parallel to the axis of the valley from a height above the valley floor approximately equal to the width of the valley. In this case the baseline lies on a circular cylinder that also lies close to the surface on which the imaged points lie. This means that it is close to one of the degenerate forms of the hyperboloid of one sheet (see Appendix C).

Note that hyperboloids of one sheet and their degenerate forms are exactly the surfaces that lead to ambiguity in the case of motion vision. The coordinate systems and symbols have been chosen here to make the correspondence between the two problems more obvious. The relationship between the two situations is nevertheless not quite as transparent as I had thought earlier [Horn 1987b]. In the case of the ambiguity of the motion field, for example, we are dealing with a two-way ambiguity arising from infinitesimal displacements. Here we are dealing with an infinite number of solutions arising from images taken with cameras that have large differences in position and orientation. Also note that the symbol $\delta\boldsymbol{\omega}$ stands for a small change in a finite rotation here, while it refers to a difference in instantaneous rotational velocities in the motion vision case.

12. Conclusions

Methods for recovering the relative orientation of two cameras are of importance in both binocular stereo and motion vision. A new iterative method for finding the relative orientation has been described here. It can be used even when there is no initial guess available for the rotation or the baseline. The new method does not use Euler angles to represent the orientation.

When there are many pairs of corresponding image points, the iterative method finds the global minimum from any starting point in parameter space. Local minima in the sum of squares of errors occur, however, when there are relatively few pairs of corresponding image points available. Method for efficiently locating the global minimum in this case were discussed. When only five pairs of corresponding image points are given, several exact solutions of the coplanarity equations can be found. Typically only one of these yields positive distances to the points in the scene, however. This allows one to pick the correct solution even when there is no initial guess available.

All of these methods fail in the rare case that the scene points lie on a critical surface.

Acknowlegements

I would like to thank W. Eric L. Grimson and Rodney A. Brooks, who made helpful comments on a draft of this paper and Shahriar Negadahripour, who worked on the application of results developed earlier for critical surfaces in motion vision in the case when the step between successive images is relatively large.

References

Bender, L.U. (1967) "Derivation of Parallax Equations," *Photogrammetric Engineering*, Vol. 33, No. 10, pp. 1175–1179, October.

Brandenberger, A. (1947) "Fehlertheorie der äusseren Orientierung von Steilaufnahmen," Ph.D. Thesis, Eidgenössische Technische Hochschule, Zürich, Switzerland.

Bruss, A.R. & B.K.P. Horn, (1983) "Passive Navigation," *Computer Vision, Graphics, and Image Processing*, Vol. 21, No. 1, January, pp. 3–20.

Brou, P. (1983) "Using the Gaussian Image to Find the Orientation of an Object," *International Journal of Robotics Research*, Vol. 3, No. 4, pp. 89–125.

Forrest, R.B. (1966) "AP-C Plotter Orientation," *Photogrammetric Engineering*, Vol. 32, No. 5, pp. 1024–1027, September.

Ghilani, C.D. (1983) "Numerically Assisted Relative Orientation of the Kern PG-2," *Photogrammetric Engineering and Remote Sensing*, Vol. 49, No. 10, pp. 1457–1459, October.

Gill, C. (1964) "Relative Orientation of Segmented, Panoramic Grid Models on the AP-II," *Photogrammetric Engineering*, Vol. 30, pp. 957–962, month?

Ghosh, S.K. (1966) "Relative Orientation Improvement," *Photogrammetric Engineering*, Vol. 32, No. 3, pp. 410–414, May.

Ghosh, S.K. (1972) *Theory of Stereophotogrammetry*, Ohio University Bookstores, Columbus, OH.

Hallert, B. (1960) *Photogrammetry*, McGraw-Hill, New York, NY.

Hilbert, D. & S. Cohn-Vossen (1953, 1983) *Geometry and the Imagination*, Chelsea Publishing, New York.

Horn, B.K.P. (1986) *Robot Vision*, MIT Press, Cambridge, MA & McGraw-Hill, New York, NY.

Horn, B.K.P. (1987a) "Closed-form Solution of Absolute Orientation using Unit Quaternions," *Journal of the Optical Society A*, Vol. 4, No. 4, pp. 629–642, April.

Horn, B.K.P. (1987b) "Motion Fields are Hardly Ever Ambiguous," *International Journal of Computer Vision*, Vol. 1, No. 3, pp. 263–278, Fall.

Horn, B.K.P. & E.J. Weldon Jr. (1988) "Direct Methods for Recovering Motion," accepted for publication by *International Journal of Computer Vision*.

Horn, B.K.P., H.M. Hilden & S. Negahdaripour (1988) "Closed-form Solution of Absolute Orientation using Orthonormal Matrices," submitted to *Journal of the Optical Society A*.

Hofmann, W. (1949) "Das Problem der 'Gefährlichen Flächen' in Theorie and Praxis," Ph.D. Thesis, Technische Hochschule München, Published in 1953 by Deutsche Geodätische Kommission, München, West Germany.

Jochmann, H. (1965) "Number of Orientation Points," *Photogrammetric Engineering*, Vol. 31. No. 4, pp. 670–679, July.

Korn, G.A. & T.M. Korn (1968) *Mathematical Handbook for Scientists and Engineers*, 2-nd edition, McGraw-Hill, New York, NY.

Longuet-Higgins, H.C. & K. Prazdny (1980) "The Interpretation of a Moving Retinal Image," *Proc. of the Royal Society of London B*, Vol. 208, pp. 385–397.

Longuet-Higgins, H.C. (1981) "A Computer Algorithm for Reconstructing a Scene from Two Projections," *Nature*, Vol. 293, pp. 133–135, September.

Longuet-Higgins, H.C. (1984) "The Reconstruction of a Scene from Two Projections—Configurations that Defeat the Eight-Point Algorithm," *IEEE, Proceedings of the First Conference on Artificial Intelligence Applications*, Denver, Colorado.

Moffit, F. & E. M. Mikhail (1980) *Photogrammetry*, 3-rd edition, Harper & Row, New York, NY.

Oswal, H.L. (1967) "Comparison of Elements of Relative Orientation," *Photogrammetric Engineering*, Vol. 33, No. 3, pp. 335–339, March.

Sailor, S. (1965) "Demonstration Board for Stereoscopic Plotter Orientation," *Photogrammetric Engineering*, Vol. 31, No. 1, pp. 176–179, January.

Salamin, E. (1979) "Application of Quaternions to Computation with Rotations," Unpublished Internal Report, Stanford University, Stanford, CA.

Schut, G.H. (1958–59) "Construction of Orthogonal Matrices and Their Application in Analytical Photogrammetry," *Photogrammetria*, Vol. 15, No. 4, pp. 149–162.

Schwidefsky, K. (1973) *An Outline of Photogrammetry*, (Translated by John Fosberry), 2-nd edition, Pitman & Sons, London, England.

Slama, C.C., C. Theurer & S.W. Henrikson, (eds.) (1980) *Manual of Photogrammetry*, American Society of Photogrammetry, Falls Church, VA.

Stuelpnagle, J.H. (1964) "On the Parametrization of the Three-Dimensional Rotation Group," *SIAM Review*, Vol. 6, No. 4, pp. 422–430, October.

Taylor, R.H. (1982) "Planning and Execution of Straight Line Manipulator Trajectories," in *Robot Motion: Planning and Control*, Brady, M.J., J.M. Hollerbach, T.L. Johnson, T. Lozano-Pérez & M.T. Mason (eds.), MIT Press, Cambridge, MA.

Thompson, E.H. (1959a) "A Method for the Construction of Orthonormal Matrices," *Photogrammetric Record*, Vol. 3, No. 13, pp. 55–59, April.

Thompson, E.H. (1959b) "A Rational Algebraic Formulation of the Problem of Relative Orientation," *Photogrammetric Record*, Vol. 3, No. 14, pp. 152–159, October.

Thompson, E.H. (1964) "A Note on Relative Orientation," *Photogrammetric Record*, Vol. 4, No. 24, pp. 483–488, October.

Thompson, E.H. (1968) "The Projective Theory of Relative Orientation," *Photogrammetria*, Vol. 23, pp. 67–75.

Tsai, R.Y. & T.S. Huang (1984) "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, No. 1, pp. 13–27, January.

Ullman, S. (1979) *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA.

Van Der Weele, A.J. (1959–60) "The Relative Orientation of Photographs of Mountainous Terrain," *Photogrammetria*, Vol. 16, No. 2, pp. 161–169.

Wolf, P.R. (1983) *Elements of Photogrammetry*, 2-nd edition, McGraw-Hill, New York, NY.

Zeller, M. (1952) *Textbook of Photogrammetry*, H.K. Lewis & Company, London, England

Appendix A—Rotation Groups of Regular Polyhedra

Each of the rotation groups of the regular polyhedra can be generated from two judiciously chosen elements. For convenience, however, an explicit representation of all of the elements of each of the groups is given here. The number of different component values occuring in the unit quaternions representing the rotations can be kept low by careful choice of the alignment of the polyhedron with the coordinate axes. The attitudes of the polyhedra here were selected to minimize the number of different numerical values that occur in the components of the unit quaternions. A different representation of the group is obtained if the vector parts of each of the unit quaternions is rotated in the same way. This just corresponds to the rotation group of the polyhedron in a different attitude with respect to the underlying coordinate system. This observation leads to a convenient way of generating finer systematic sampling patterns of the space of rotations than the ones provided directly by the rotation group of a regular polyhedron in a particular alignment with the coordinate axes (see also [Brou 1983]).

The components of the unit quaternions here may take on the values 0 and 1, as well as the following:

$$a = \frac{\sqrt{5}-1}{4}, \quad b = \frac{1}{2}, \quad c = \frac{1}{\sqrt{2}}, \quad \text{and} \quad d = \frac{\sqrt{5}+1}{4}.$$

Here are the unit quaternions for the twelve elements of the rotation group of the tetrahedron:

$(1, 0, 0, 0)$	$(0, 1, 0, 0)$	$(0, 0, 1, 0)$	$(0, 0, 0, 1)$
(b, b, b, b)	$(b, b, b, -b)$	$(b, b, -b, b)$	$(b, b, -b, -b)$
$(b, -b, b, b)$	$(b, -b, b, -b)$	$(b, -b, -b, b)$	$(b, -b, -b, -b)$

Here are the unit quaternions for the twenty-four elements of the rotation group of the octahedron and the hexahedron (cube):

$(1, 0, 0, 0)$	$(0, 1, 0, 0)$	$(0, 0, 1, 0)$	$(0, 0, 0, 1)$
$(0, 0, c, c)$	$(0, 0, c, -c)$	$(0, c, 0, c)$	$(0, c, 0, -c)$
$(0, c, c, 0)$	$(0, c, -c, 0)$	$(c, 0, 0, c)$	$(c, 0, 0, -c)$
$(c, 0, c, 0)$	$(c, 0, -c, 0)$	$(c, c, 0, 0)$	$(c, -c, 0, 0)$
(b, b, b, b)	$(b, b, b, -b)$	$(b, b, -b, b)$	$(b, b, -b, -b)$
$(b, -b, b, b)$	$(b, -b, b, -b)$	$(b, -b, -b, b)$	$(b, -b, -b, -b)$

Here are the unit quaternions for the sixty elements of the rotation group of the icosahedon and the dodecahedron:

(1, 0, 0, 0)	(0, 1, 0, 0)	(0, 0, 1, 0)	(0, 0, 0, 1)
(0, a, b, d)	(0, a, b,-d)	(0, a,-b, d)	(0, a,-b,-d)
(0, b, d, a)	(0, b, d,-a)	(0, b,-d, a)	(0, b,-d,-a)
(0, d, a, b)	(0, d, a,-b)	(0, d,-a, b)	(0, d,-a,-b)
(a, 0, d, b)	(a, 0, d,-b)	(a, 0,-d, b)	(a, 0,-d,-b)
(b, 0, a, d)	(b, 0, a,-d)	(b, 0,-a, d)	(b, 0,-a,-d)
(d, 0, b, a)	(d, 0, b,-a)	(d, 0,-b, a)	(d, 0,-b,-a)
(a, b, 0, d)	(a, b, 0,-d)	(a,-b, 0, d)	(a,-b, 0,-d)
(b, d, 0, a)	(b, d, 0,-a)	(b,-d, 0, a)	(b,-d, 0,-a)
(d, a, 0, b)	(d, a, 0,-b)	(d,-a, 0, b)	(d,-a, 0,-b)
(a, d, b, 0)	(a, d,-b, 0)	(a,-d, b, 0)	(a,-d,-b, 0)
(b, a, d, 0)	(b, a,-d, 0)	(b,-a, d, 0)	(b,-a,-d, 0)
(d, b, a, 0)	(d, b,-a, 0)	(d,-b, a, 0)	(d,-b,-a, 0)
(b, b, b, b)	(b, b, b,-b)	(b, b,-b, b)	(b, b,-b,-b)
(b,-b, b, b)	(b,-b, b,-b)	(b,-b,-b, b)	(b,-b,-b,-b)

Remember that changing the signs of all the components of a unit quaternion does not change the rotation that it represents.

Appendix B—Some Properties of Critical Surfaces

In this appendix we develop some more of the properties of the critical surfaces. The equation of a critical surface can be written in the form

$$(\mathbf{R} \times \mathbf{b}) \cdot (\delta\boldsymbol{\omega} \times \mathbf{R}) + \mathbf{L} \cdot \mathbf{R} = 0,$$

or

$$(\mathbf{R} \cdot \mathbf{b})(\delta\boldsymbol{\omega} \cdot \mathbf{R}) - (\mathbf{b} \cdot \delta\boldsymbol{\omega})(\mathbf{R} \cdot \mathbf{R}) + \mathbf{L} \cdot \mathbf{R} = 0,$$

where

$$\mathbf{L} = \boldsymbol{\ell} \times \mathbf{b}, \quad \text{while} \quad \boldsymbol{\ell} = \mathbf{b} \times \delta\boldsymbol{\omega} + \delta\mathbf{b}.$$

It is helpful to first establish some simple relationships between the quantities appearing in the formula above. We start with the observations that $\boldsymbol{\ell} \cdot \mathbf{b} = 0$, that $\boldsymbol{\ell} \cdot \delta\boldsymbol{\omega} = \delta\mathbf{b} \cdot \delta\boldsymbol{\omega}$, and $\boldsymbol{\ell} \times \delta\mathbf{b} = -(\delta\mathbf{b} \cdot \delta\boldsymbol{\omega})\mathbf{b}$.

We can also expand $\mathbf{L}$ to yield,

$$\mathbf{L} = \delta\boldsymbol{\omega} - (\mathbf{b} \cdot \delta\boldsymbol{\omega})\mathbf{b} + \delta\mathbf{b} \times \mathbf{b}.$$

It follows that $\mathbf{L} \cdot \mathbf{b} = 0$, that $\mathbf{L} \cdot \delta\mathbf{b} = \delta\boldsymbol{\omega} \cdot \delta\mathbf{b}$, and

$$\mathbf{L} \times \mathbf{b} = -(\mathbf{b} \times \delta\boldsymbol{\omega} + \delta\mathbf{b}) = -\boldsymbol{\ell},$$

$$\mathbf{L} \times \delta\boldsymbol{\omega} = (\delta\mathbf{b} \cdot \delta\boldsymbol{\omega})\mathbf{b} - (\mathbf{b} \cdot \delta\boldsymbol{\omega})\boldsymbol{\ell}.$$

We have already established that $\mathbf{R} = k\,\mathbf{b}$ is an equation for one of the rulings passing through the origin. A hyperboloid of one sheet has two intersecting families of rulings, so there should be a second ruling passing through the origin. Consider the vector $\mathbf{S}$ defined by

$$\mathbf{S} = (\mathbf{L} \times \delta\boldsymbol{\omega}) \times \mathbf{L},$$

which can be written in the form

$$\mathbf{S} = (\mathbf{L} \cdot \mathbf{L})\delta\boldsymbol{\omega} - (\mathbf{L} \cdot \delta\boldsymbol{\omega})\mathbf{L},$$

or

$$\mathbf{S} = (\delta\boldsymbol{\omega} \cdot \delta\mathbf{b})\boldsymbol{\ell} + (\mathbf{b} \cdot \delta\boldsymbol{\omega})(\boldsymbol{\ell} \cdot \boldsymbol{\ell})\mathbf{b},$$

so that $\mathbf{S} \cdot \mathbf{b} = (\mathbf{b} \cdot \delta\boldsymbol{\omega})(\boldsymbol{\ell} \cdot \boldsymbol{\ell})$ and $\mathbf{S} \cdot \delta\boldsymbol{\omega} = (\delta\boldsymbol{\omega} \cdot \delta\mathbf{b})^2 + (\mathbf{b} \cdot \delta\boldsymbol{\omega})^2(\boldsymbol{\ell} \cdot \boldsymbol{\ell})$.

If we substitute $\mathbf{R} = k\,\mathbf{S}$ into the formula

$$(\mathbf{R} \times \mathbf{b}) \cdot (\delta\boldsymbol{\omega} \times \mathbf{R}) + \mathbf{L} \cdot \mathbf{R},$$

we obtain zero, since $\mathbf{L} \cdot \mathbf{S} = 0$ and

$$\mathbf{S} \times \mathbf{b} = (\delta\boldsymbol{\omega} \cdot \delta\mathbf{b})\mathbf{L}$$

is orthogonal to

$$\mathbf{S} \times \delta\boldsymbol{\omega} = -(\mathbf{L} \cdot \delta\boldsymbol{\omega})\,\mathbf{L} \times \delta\boldsymbol{\omega}.$$

We conclude that $\mathbf{R} = k\mathbf{S}$ is an equation for the other ruling that passes through the right projection center.

There are two families of parallel planes that cut an ellipsoid in circular cross-sections [Hilbert & Cohn-Vossen 1953]. Similarly, there are two families of parallel planes that cut a hyperboloid of one sheet in circular cross-sections. One of these families consists of planes perpendicular to the baseline, that is, with common normal $\mathbf{b}$. We can see this by substituting $\mathbf{R} \cdot \mathbf{b} = k$ in the equation of the critical surface. We obtain

$$k(\delta\boldsymbol{\omega} \cdot \mathbf{R}) - (\mathbf{b} \cdot \delta\boldsymbol{\omega})(\mathbf{R} \cdot \mathbf{R}) + \mathbf{L} \cdot \mathbf{R} = 0,$$

or

$$(\mathbf{b} \cdot \delta\boldsymbol{\omega})(\mathbf{R} \cdot \mathbf{R}) - (k\,\delta\boldsymbol{\omega} + \mathbf{L}) \cdot \mathbf{R} = 0.$$

This is the equation of a sphere, since the only second-order term in $\mathbf{R}$ is a multiple of

$$\mathbf{R} \cdot \mathbf{R} = X^2 + Y^2 + Z^2.$$

We can conclude that the intersection of the critical surface and the plane is also the intersection of this sphere and the plane, and so must be a circle. The same applies to the intersection of the critical surface and the family of planes with common normal $\delta\boldsymbol{\omega}$, since we get

$$(\mathbf{b} \cdot \delta\boldsymbol{\omega})(\mathbf{R} \cdot \mathbf{R}) - (k\,\mathbf{b} + \mathbf{L}) \cdot \mathbf{R} = 0,$$

when we substitute $\mathbf{R} \cdot \delta\boldsymbol{\omega} = k$ into the equation of the critical surface.

The equation of the critical surface is given in the implicit form $f(\mathbf{R}) = 0$. The equation of a tangent plane to such a surface can be obtained by differentiating with respect to $\mathbf{R}$:

$$\mathbf{N} = (\mathbf{R} \times \delta\boldsymbol{\omega}) \times \mathbf{b} + (\mathbf{R} \times \mathbf{b}) \times \delta\boldsymbol{\omega} + \mathbf{L}$$

or

$$\mathbf{N} = (\mathbf{R}\cdot\mathbf{b})\delta\boldsymbol{\omega} + (\mathbf{R}\cdot\delta\boldsymbol{\omega})\mathbf{b} - 2(\mathbf{b}\cdot\delta\boldsymbol{\omega})\mathbf{R} + \mathbf{L}$$

The tangent plane at the origin has normal $\mathbf{L}$. This tangent plane contains the baseline (since $\mathbf{L}\cdot\mathbf{b} = 0$), as well as the other ruling passing through the origin (since $\mathbf{L}\cdot\mathbf{S} = 0$). Note that the normal to the tangent plane is not constant along either of these rulings, as they would be if we were dealing with a developable surface.

In the above we have not considered a large number of degenerate situations that can occur. The reader is referred to Appendix C for a detailed analysis of these.

Appendix C—Degenerate Critical Surfaces

There are a number of special alignments of the infinitesimal change in the rotation, $\delta\boldsymbol{\omega}$, with the baseline, $\mathbf{b}$, and the infinitesimal change in the baseline $\delta\mathbf{b}$ that lead to degenerate forms of the hyperboloid of one sheet.

One of the rulings passing through the origin is given by $\mathbf{R} = k\,\mathbf{b}$, while the other is given by $\mathbf{R} = k\,\mathbf{S}$. If these two rulings become parallel, we are dealing with a degenerate form that has only one set of rulings, that is a conical surface. Now

$$\mathbf{S} = (\delta\boldsymbol{\omega}\cdot\delta\mathbf{b})\boldsymbol{\ell} + (\mathbf{b}\cdot\delta\boldsymbol{\omega})(\boldsymbol{\ell}\cdot\boldsymbol{\ell})\mathbf{b},$$

is parallel to $\mathbf{b}$ only when $(\delta\boldsymbol{\omega}\cdot\delta\mathbf{b}) = 0$, since $\boldsymbol{\ell}$ is perpendicular to $\mathbf{b}$. In this case

$$\delta\mathbf{b}\cdot\delta\boldsymbol{\omega} = 0 \quad \text{and} \quad \delta\mathbf{b}\cdot\mathbf{b} = 0,$$

so $\delta\mathbf{b} = k(\mathbf{b}\times\delta\boldsymbol{\omega})$ for some constant k. Consequently $\boldsymbol{\ell} = (k+1)(\mathbf{b}\times\delta\boldsymbol{\omega})$, The vertex of the conical surface must lie on the baseline since the baseline is a ruling, and every ruling passes through the vertex. It can be shown that the vertex actually lies at $\mathbf{R} = -(k+1)\mathbf{b}$.

We also know that cross-sections in planes perpendicular to the baseline are circles. This tells us that we are dealing with elliptical cones. Right circular cones cannot be critical surfaces. It can be shown that the main axis of the elliptical cone lies in the direction $\mathbf{b} + \delta\boldsymbol{\omega}$.

A special case of the special case above occurs when

$$\|\mathbf{b}\times\delta\boldsymbol{\omega}\| = 0,$$

that is $\delta\boldsymbol{\omega} \parallel \mathbf{b}$. Here $\delta\boldsymbol{\omega} = k\,\mathbf{b}$ for some constant k and so $\boldsymbol{\ell} = \delta\mathbf{b}$ and $\mathbf{L} = \delta\mathbf{b}\times\mathbf{b}$. The equation of the surface becomes

$$k(\mathbf{R}\cdot\mathbf{b})^2 - k(\mathbf{b}\cdot\mathbf{b})(\mathbf{R}\cdot\mathbf{R}) + \mathbf{L}\cdot\mathbf{R} = 0,$$

or

$$k\,\|\mathbf{R}\times\mathbf{b}\|^2 + (\delta\mathbf{b}\times\mathbf{b})\cdot\mathbf{R} = 0.$$

This is the equation of a circular cylinder with axis parallel to the baseline. In essence, the vertex of the cone has receded to infinity along the baseline.

Another special case arises when the radius of the circular cross-sections with planes perpendicular to the baseline becomes infinite. In this case we obtain straight lines, and hence rulings, in these planes. The hyperbolic paraboloid is the degenerate form that has the property that each of its two sets of rulings can be obtained by cutting the surface with a set of parallel planes [Hilbert & Cohn-Vossen 1953]. This happens when $\delta\boldsymbol{\omega}$ is perpendicular to $\mathbf{b}$, that is, $\mathbf{b} \cdot \delta\boldsymbol{\omega} = 0$. The equation of the surface in this case simplifies to

$$(\mathbf{R} \cdot \mathbf{b})(\delta\boldsymbol{\omega} \cdot \mathbf{R}) + \mathbf{L} \cdot \mathbf{R} = 0.$$

The intersection of this surface with any plane perpendicular to the baseline is a straight line. We can show this by substituting

$$\mathbf{R} \cdot \mathbf{b} = k,$$

into the equation of the surface. We obtain

$$(k\, \delta\boldsymbol{\omega} + \mathbf{L}) \cdot \mathbf{R} = 0,$$

that is, the equation of another plane. Now the intersection of two planes is a straight line. So we may conclude that the intersection of the surface and the original plane is a straight line. We can show in the same way that the intersection of the surface with any plane perpendicular to $\delta\boldsymbol{\omega}$ is a straight line by substituting

$$\mathbf{R} \cdot \delta\boldsymbol{\omega} = k$$

into the equation of the surface. It can be shown that the saddle point of the hyperbolic paraboloid surface lies on the baseline.

A special case of particular interest arises when $\delta\boldsymbol{\omega}$ is perpendicular to both $\mathbf{b}$ and $\delta\mathbf{b}$, that is,

$$\mathbf{b} \cdot \delta\boldsymbol{\omega} = 0 \quad \text{and} \quad \delta\mathbf{b} \cdot \delta\boldsymbol{\omega} = 0,$$

and so $\delta\boldsymbol{\omega} = k(\delta\mathbf{b} \times \mathbf{b})$, for some constant k. Then $\boldsymbol{\ell} = (k+1)\,\delta\mathbf{b}$ and $\mathbf{L} = (k+1)(\delta\mathbf{b} \times \mathbf{b})$. The equation of the surface becomes

$$k(\mathbf{R} \cdot \mathbf{b})\big((\delta\mathbf{b} \times \mathbf{b}) \cdot \mathbf{R}\big) + (k+1)\big((\delta\mathbf{b} \times \mathbf{b}) \cdot \mathbf{R}\big) = 0,$$

or just

$$\big(k(\mathbf{R} \cdot \mathbf{b}) + (k+1)\big)\big((\delta\mathbf{b} \times \mathbf{b}) \cdot \mathbf{R}\big) = 0.$$

so either

$$(\delta\mathbf{b} \times \mathbf{b}) \cdot \mathbf{R} = 0 \quad \text{or} \quad k(\mathbf{R} \cdot \mathbf{b}) + (k+1) = 0.$$

The first of these is the equation of a plane containing the baseline $\mathbf{b}$ and the vector $\delta\mathbf{b}$. The second is the equation of a plane perpendicular to the baseline. So the solution degenerates in this case into a surface consisting of two intersecting planes. One of these planes appears only as a line in each of the two images, since it passes through both projection centers, and so does not really contribute to the image. It is fortunate that planes can only be degenerate surfaces if they are perpendicular to the baseline, since surfaces that are almost planar occur frequently in aerial photography.[1]

To summarize then, we have the following degenerate cases:

- elliptical cones when $\delta\boldsymbol{\omega} \perp \delta\mathbf{b}$,
- circular cylinders when $\delta\boldsymbol{\omega} \parallel \mathbf{b}$,
- hyperbolic parabloids when $\delta\boldsymbol{\omega} \perp \mathbf{b}$,
- intersecting planes when $\delta\boldsymbol{\omega} \perp \delta\mathbf{b}$ and $\delta\boldsymbol{\omega} \perp \mathbf{b}$.

For further details, and a proof that not all hyperboloids of one sheet passing through the origin lead to critical surfaces, see [Horn 1987b].

[1]The baseline was nearly perpendicular to the surface in the sequence of photographs obtained by the Ranger spacecraft as it crashed into the lunar surface. This made photogrammetric analysis difficult.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AI Memo 994	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD-A190385
4. TITLE (and Subtitle) Relative Orientation		5. TYPE OF REPORT & PERIOD COVERED memorandum
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Berthold K.P. Horn		8. CONTRACT OR GRANT NUMBER(s) DACA76-85-C-0100 N00014-85-K-0124
9. PERFORMING ORGANIZATION NAME AND ADDRESS Artificial Intelligence Laboratory 545 Technology Square Cambridge, MA 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, VA 22209		12. REPORT DATE September 1987
		13. NUMBER OF PAGES 31
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Office of Naval Research Information Systems Arlington, VA 22217		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

relative orientation — photogrammetry
binocular stereo — motion vision
coplanarity condition — representation of rotation

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Before corresponding points in images taken with two cameras can be used to recover distances to objects in a scene, one has to determine the position and orientation of one camera relative to the other. This is the classic photogrammetric problem of "relative orientation," central to the interpretation of binocular stereo information. Iterative methods for determining relative orientation were developed long ago; without them we would not have most of the topographic maps we do today. Relative orientation is also of importance

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Completed 7-28-88

Block 20 cont.

in the recovery of motion and shape from an image sequence when successive frames are widely separated in time. Workers in motion vision are rediscovering some of the methods of photogrammetry.

Described here is a particularly simple iterative scheme for recovering relative orientation that, unlike existing methods, does not require a good initial guess for the baseline and the rotation. The data required is a set of pairs of corresponding rays from the two projection centers to points in the scene. It is well known that at least five pairs of rays are needed. Less appears to be known about the existence of multiple solutions and their interpretation. These issues are discussed here in detail. The unambiguous determination of all of the parameters of relative orientation is not possible when the observed points lie on a critical surface. These surfaces and their degenerate forms are analysed here as well.