DECOMPOSITION OF TRANSFORMATION MATRICES FOR ROBOT VISION

Sundaram Ganapathy

Robotics Systems Research Department, 4E-618

AT&T Bell Laboratories, Holmdel, New Jersey, 07733

Abstract

The relationship between the three-dimensional coordinates of a point and the corresponding two-dimensional coordinates of its image, as seen by a camera, can be expressed in terms of a 3 by 4 matrix using the homogeneous coordinate system. This matrix is known more generally as the transformation matrix and can be determined experimentally by measuring the image coordinates of six or more known points in space. Such a transformation can also be derived analytically from knowledge of the camera position, orientation, focal length and scaling and translation parameters in the image plane. However, the inverse problem of computing the camera location and orientation from the transformation matrix involves solution of simultaneous nonlinear equations in several variables and is considered difficult.

In this paper we present a new and simple analytical technique that accomplishes this inversion rather easily. This technique works very well in practice and has considerable applications for motion tracking.

1.0 INTRODUCTION

It is well known that the two dimensional image coordinates of the image of a point in space, as seen by a camera, can be conveniently expressed in terms of a 3 by 4 matrix using the homogeneous coordinate system. Roberts [1] provides a good treatment of the homogeneous coordinate system and its use in obtaining this transformation analytically. Such a transformation is variously referred to in the literature as camera calibration matrix [2] or perspective transformation matrix [3] or more simply as the transformation matrix. Properties of the transformation matrix and techniques for deriving this matrix can also be found in Duda and Hart [4] and Haralick [3] . Haralick [3] provides a good review of the properties and the uses of this matrix for several reconstruction problems in computer vision. This transformation matrix can also be determined experimentally by observing the images of six or more known points in space and doing a least-square fit solution for the resultant system of overly constrained linear simultaneous equations. Sutherland [5] provides the actual equations and a method of solution. Sobel [2] discusses the problems and errors involved in the experimental determination of a camera calibration matrix and proposes a method of improving accuracy of calibration.

This matrix has considerable applications in the fields of computer graphics and computer vision. Rogers and Adams [6] is a good source for applications in computer graphics and Ballard and Brown [7] for applications in computer vision. The determination of this matrix is the starting point for several problems in the area of computer vision - in particular stereo reconstruction [8, 9, 10], guiding unmanned vehicles [11], dynamic scene analysis [12, 13], to name a few. In all these applications, determination of the camera location and orientation from measurements on two dimensional images is very useful and sometimes necessary. Yet, even though the problems of camera calibration and determination of this matrix has received considerable attention in the literature, the problem of camera location determination has been given relatively little attention except in the field of photogrammetry where Church's method [14] for determination of camera location and orientation from knowledge of the focal length of the camera and three points in space and their corresponding images is well known [15]. However, Fischler and Bolles [16] claim that little is known about the conditions under which one can obtain unique solutions and the minimum number of image points needed for unique solutions. Subsequently, Ganapathy [17] provided a complete treatment of the minimum number of points needed for unique solutions for several classes of problems involving partial/no knowledge of camera parameters.

Here we are concerned only with the inverse problem - namely given a transformation matrix, determine the camera location, orientation, scaling and translation parameters in the image plane from it. Some research has been done in this area by Binford and his colleagues at Stanford university and the reader is referred to papers by Gennery [10] and Lowe [18] for their approach to this problem. This inverse problem is equivalent to the following problem: given a photograph that has been enlarged both horizontally and vertically (possibly different scales in the two directions) and clipped arbitrarily, determine the position and orientation of the camera used to take the picture from knowledge of the correspondences between known objects and their images in the photograph. Any formulation of this problem results in a set of nonlinear

CH2008-1/84/0000/0130$01.00©1984 IEEE

simultaneous equations in several (at least six, depending on the formulation) independent variables and hence the methods that have been proposed are iterative in nature and use hill-climbing or other well known numerical techniques for solution. Closed form solution to this problem is considered difficult and I quote (from pp. 123 [18]) "However, because of problems in calculating a rotation in terms of its three underlying parameters, there appears to be no straightforward symbolic solution to the problem, and we are forced to seek an iterative solution".

In this paper we derive a new and simple non-iterative analytical technique that solves the camera location determination problem. However, in order to do that, it is necessary to represent the transformation matrix in a different way than is conventional and we derive such a representation in the next section. With this new representation, the inverse problem can be stated succinctly. We present the problem statement in section 3 and briefly describe the rationale behind the new technique. The section also addresses some misconceptions regarding the degrees of freedom in the specification of a transformation matrix. The properties of a pure 3 by 3 rotation matrix play an important role in the analytical solution and hence they are covered in section 4. With these developments behind us, we are ready for the solution which is surprisingly simple! This is presented in section 5 for an ideal transformation matrix with no errors in it. In section 6, we present a modified version of the same algorithm to handle a real transformation matrix with possible errors in it. Finally, in Appendix B, we illustrate our solution technique on a transformation matrix, obtained experimentally [19].

2.0 DERIVATION OF THE TRANSFORM MATRIX

Familiarity with the homogeneous coordinate system is necessary to understand the details of the derivation. In such a system a three-dimensional point x, y, z is represented by a 4-tuple wx, wy, wz, w by adding an extra scalar w. To obtain the three-dimensional coordinates of a point represented in the homogeneous coordinate system one merely divides by the last component of the 4-tuple. Similarly, a two-dimensional point u, v is represented by wu, wv, w and we divide by the third component w to obtain u and v. This representation is useful in obtaining perspective transformations and the reader is referred elsewhere for more details [1, 6].

Consider a camera center S located at X_c, Y_c, Z_c (measured in the X, Y, Z coordinate system) looking along a line of sight $SO'P$. P is the point at which the line of sight pierces the $X-Y$ plane and O' is the point at which the line of sight intersects the image plane (see figure 1). Let $S'P$ represent the projection of SP on the $X-Y$ plane. We can transform the x, y, z coordinates of a point to x', y', z' coordinates by choosing an $X'Y'Z'$ system of axes centered at S in three steps as follows:

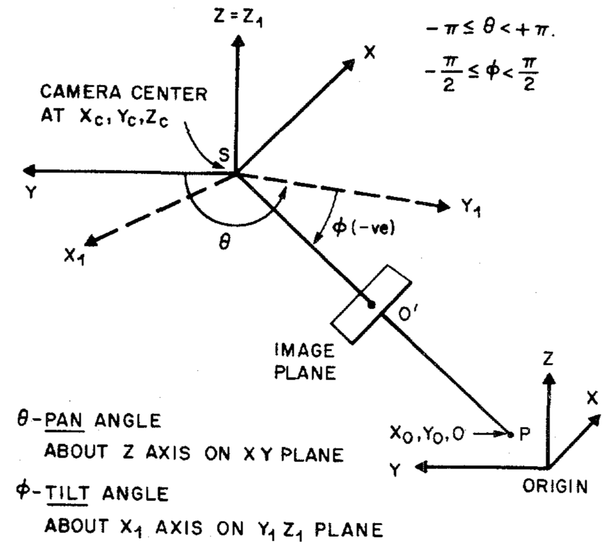


FIGURE 1 – CAMERA GEOMETRY

2.1 STEP1 (MOVE ORIGIN)

First move the origin from O to S leaving the axes the same. This can be accomplished with a displacement matrix D and can be represented as a 4 by 4 matrix in the homogeneous coordinate system. D is given by

$$D = \begin{bmatrix} 1 & 0 & 0 & -X_c \\ 0 & 1 & 0 & -Y_c \\ 0 & 0 & 1 & -Z_c \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.2 STEP2 (PAN)

Rotate the $X-Y$ plane around the Z axis by an angle θ such that the new Y axis is parallel to $S'P$ and is in the same direction as $S'P$ (see figure 1). Note that θ can be between $-\pi$ and $+\pi$. If we take the positive direction of rotation in the conventional sense namely X to Y the transformation matrix (say $R1$) is

$$R1 = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.3 STEP3 (TILT)

Step 2 gives us a new X axis, a new Y axis and leaves the Z axis intact. Now rotate the new $Y-Z$ plane around the new X axis by an angle ϕ (see figure 1) to align the Y axis with SP. The positive direction of rotation is once again from Y to Z as shown in the figure. It is important to note that ϕ is between $-\frac{\pi}{2}$ and $+\frac{\pi}{2}$. This will be used later in the decomposition process. The transformation matrix $R2$ that does this is

$$R2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & \sin\phi & 0 \\ 0 & -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2.4 STEP4 (SWING)

Now we obtain a camera centered coordinate system with the new Y axis aligned along the line of sight and pointing towards the image plane at a distance F from the camera center S. The projection of the X' axis and the Z' axis on the image plane is shown in figure 2. However the image coordinates of an image point may be measured with respect to $U-V$ axes (direction unknown) centered at the image origin I (location unknown) as shown in figure 2. We can transform the $X'-Z'$ axes by rotating around the Y' axis by an angle ψ such that the new X' axis is aligned parallel to the U axis. This will leave the new Z' axis either aligned in the same direction as the V axis or parallel to it but in the opposite direction. It is important to note however that ψ can be chosen so that the new X' axis is in the same direction as the U axis. The matrix $R3$ that accomplishes this is

$$R3 = \begin{bmatrix} \cos\psi & 0 & -\sin\psi & 0 \\ 0 & 1 & 0 & 0 \\ \sin\psi & 0 & \cos\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

We will name the resultant coordinate system the image centered coordinate system. The image plane itself is at $y' = F$. The coordinates x', y', z' of a point x, y, z (in world coordinates) is given by

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} EXT \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

where EXT is a 4 by 4 transform matrix given by

$$EXT = [R3]\,[R2]\,[R1]\,[D] = [R]\,[D]$$

If we write R as

$$\begin{bmatrix} R \end{bmatrix} = \begin{bmatrix} a & b & c & 0 \\ d & e & f & 0 \\ g & h & i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

then

$$\begin{bmatrix} EXT \end{bmatrix} = \begin{bmatrix} a & b & c & p \\ d & e & f & q \\ g & h & i & r \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where the nine parameters a, b, c, d, e, f, g, h and i are given by

$$a = \cos\psi\cos\theta - \sin\theta\sin\phi\sin\psi \qquad (2.1)$$
$$b = \cos\psi\sin\theta + \cos\theta\sin\phi\sin\psi \qquad (2.2)$$
$$c = -\cos\phi\sin\psi \qquad (2.3)$$
$$d = -\sin\theta\cos\phi \qquad (2.4)$$
$$e = \cos\theta\cos\phi \qquad (2.5)$$
$$f = \sin\phi \qquad (2.6)$$
$$g = \cos\theta\sin\psi + \sin\theta\sin\phi\cos\psi \qquad (2.7)$$
$$h = \sin\theta\sin\psi - \cos\theta\sin\phi\cos\psi \qquad (2.8)$$
$$i = \cos\phi\cos\psi \qquad (2.9)$$

and p, q, r are expressed in terms of $a, b, c, d, e, f, g, h, i$ as

$$p = -aX_c - bY_c - cZ_c \qquad (2.10a)$$
$$q = -dX_c - eY_c - fZ_c \qquad (2.10b)$$
$$r = -gX_c - hY_c - iZ_c \qquad (2.10c)$$

Now to convert the camera-centered x', y', z' coordinates to image-centered u, v coordinates on the image plane we take the following steps (step 5 through 7)

2.5 STEP 5 (PERSPECTIVE)

Obtain the x'', z'' coordinates of an image point by the perspective transformation as follows

$$x'' = \frac{x'F}{y'} \quad \text{and} \quad z'' = \frac{z'F}{y'}$$

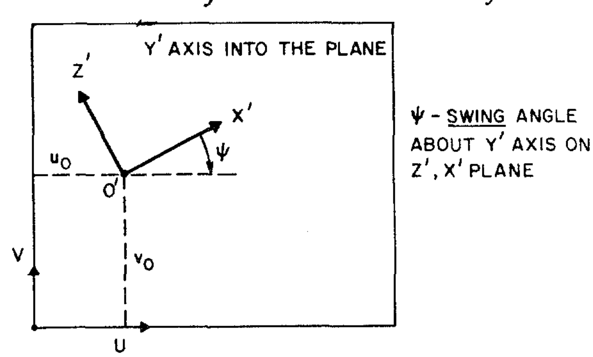
2.6 STEP 6 (RASTERIZATION)

To convert the measurement units to image units (to rasterize for example), scale by k_u in the X'' axis and k_v in the Z'' axis. Note that k_u is always positive. k_v may be positive or negative.

2.7 STEP 7 (MOVE IMAGE ORIGIN)

Translate the origin to I. If the coordinates of O' are u_0 and v_0 in the $U-V$ system then combining steps 5 through 7 we get

$$u = u_0 + \frac{k_u x' F}{y'} \quad \text{and} \quad v = v_0 + \frac{k_v z' F}{y'}$$



$$u = u_0 + k_u \frac{F}{y'} x' \qquad \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_1 & u_0 & 0 \\ 0 & v_0 & k_2 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix}$$
$$v = v_0 + k_v \frac{F}{y'} z'$$

NOTE $k_1 = k_u F$ AND IS + ve
$k_2 = k_v F$ AND COULD BE + ve OR - ve
AND $0 \le \psi < 2\pi$

FIGURE 2 - IMAGE PLANE GEOMETRY

Using the homogeneous coordinate system this can be written as

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} INT \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}$$

where *INT* is a 3 by 4 matrix whose components are

$$\begin{bmatrix} k_u F & u_0 & 0 & 0 \\ 0 & v_0 & k_v F & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Note that if we denote k_uF by k_1 and k_vF by k_2, in the 3 by 4 matrix *INT* there is no reference to F at all. In fact the focal length F is not recoverable from any measurements of image coordinates. We will show that k_1 and k_2 are indeed recoverable but F by itself is not, nor are k_u and k_v. Thus we can represent the whole imaging process by a 3 by 4 matrix T known as the transformation matrix which expresses the relationship between image coordinates and world coordinates as follows

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} T \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

and

$$[T] = [INT] \quad [EXT]$$

INT is expressed in terms of 4 parameters k_1, k_2, u_0 and v_0 and *EXT* is expressed in terms of the six parameters X_c, Y_c, Z_c, θ, ϕ and ψ.

3.0 DECOMPOSING THE TRANSFORM MATRIX

Let us represent the components of the transform matrix T by t_{11}, t_{12} .... t_{33}, t_{34}. These can be also be expressed as

$$\begin{aligned}
t_{11} &= k_1 a + u_0 d \\
t_{12} &= k_1 b + u_0 e \\
t_{13} &= k_1 c + u_0 f \\
t_{14} &= k_1 p + u_0 q \\
t_{21} &= k_2 g + v_0 d \\
t_{22} &= k_2 h + v_0 e \\
t_{23} &= k_2 i + v_0 f \\
t_{24} &= k_2 r + v_0 q \\
t_{31} &= d \\
t_{32} &= e \\
t_{33} &= f \\
t_{34} &= q
\end{aligned}$$

Note that if all terms of the transformation matrix T were scaled by the same amount (say dividing by q), no change occurs in image coordinates because of the use of the homogeneous coordinate system and the subsequent division by the third row to get u and v. Thus it can be given in terms of eleven t_{ij} with t_{34} set to 1. The problem we wish to address involves decomposing this matrix to arrive at k_1, k_2, u_0, v_0, X_c, Y_c, Z_c, θ, ϕ and ψ. One way to look at it is that we are given eleven equations expressing eleven t_{ij} in terms of sixteen unknowns k_1, k_2, u_0, v_0, X_c, Y_c, Z_c, a, b, c, d, e, f, g, h and i. However, it is easily seen that the nine unknowns a, b, c, d, e, f, g, h and i can be specified in terms of the three unknowns θ, ϕ and ψ. Therefore we really have eleven equations in only ten unknowns(!) which means we have an overconstrained system of equations. However expressing a, b, c, d, e, f, g, h and i in terms of θ, ϕ and ψ is not only messy but makes the decomposition algorithm harder to understand. Hence we will take the following approach: We will leave it in terms of sixteen unknowns but impose additional (6) constraints to force the nine components a through i to form a 3 by 3 pure rotation matrix.

It is worth noting that there are only ten degrees of freedom associated with a perspective transformation matrix. The six degrees of freedom associated with camera location (3) and camera orientation (3) are obvious. The other four consist of two for scaling and two for location of origin in the image plane. If we measure the image coordinates directly on the image plane then the two degrees of freedom associated with scaling do not exist. Further if we assume the origin of measurement is O' (the principal point on the image plane) then the degrees of freedom are reduced by two more to a total of six. Thus, in principle, we can solve for camera location and orientation from knowledge of three points and their corresponding images. In such a case we obtain six equations, two for each correspondence. However, since the resultant equations are polynomials of at least the second degree, in general we get multiple solutions. But it so happens that if we know four points on a plane and their corresponding images (as measured directly on the image plane with no scaling) then we can uniquely locate the camera position and orientation.

In this paper we are interested in a general solution with no restrictions. Since we have ten independent unknowns, we need ten independent equations. It is generally believed [5, 7, 16 etc] that a minimum of six points and their corresponding images are needed to determine the transformation matrix T uniquely. The argument made is that there are eleven independent components in the matrix T and hence that we need at least eleven equations. It is true that there are eleven components in the transformation matrix T but not all eleven can be independent because, as we have shown, the eleven can be expressed in turn using a maximum of only ten independent variables. Thus given an experimentally obtained transformation matrix, because of errors in the terms t_{ij} of the matrix, we can obtain different solutions depending on which ten of the t_{ij} we choose to use. Hence we will first develop an analytical technique of solution for the case with no errors in the terms t_{ij} and then modify it for application in practice. In the next section we describe the properties of a pure rotation matrix which are used in the development of the algorithm.

4.0 PROPERTIES OF THE ROTATION MATRIX

Let R be a 3 by 3 rotation matrix expressed as

$$R = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

A pure rotation matrix R must be a "proper orthonormal matrix" [20]. Hence it follows that R^{-1} is R^T and is given by

$$R^{-1} = \begin{bmatrix} a & d & g \\ b & e & h \\ c & f & i \end{bmatrix}$$

This property along with the property that the determinant of R must be unity implies the following constraints

$$a^2 + b^2 + c^2 = d^2 + e^2 + f^2 = g^2 + h^2 + i^2 = 1$$

$$i = ae - bd \qquad g = bf - ce \qquad h = cd - af$$

$$a = ei - hf \qquad b = fg - di \qquad c = dh - eg$$

$$d = hc - bi \qquad e = ai - gc \qquad f = bg - ah$$

$$ad + be + cf = dg + eh + fi = ag + bh + ci = 0$$

Obviously not all of these are independent. The interdependence can be recognized from the following algebraic identity

$$(ad+be+cf)^2 = (a^2+b^2+c^2)(d^2+e^2+f^2) - (ae-bd)^2 - (cd-af)^2 - (bf-ec)^2$$

Thus if we choose the 6 constraints to be

$$a^2 + b^2 + c^2 = 1 \qquad \text{(C1)}$$
$$d^2 + e^2 + f^2 = 1 \qquad \text{(C2)}$$
$$g^2 + h^2 + i^2 = 1 \qquad \text{(C3)}$$
$$i = ae - bd \qquad \text{(C4)}$$
$$h = cd - af \qquad \text{(C5)}$$
$$g = bf - ec \qquad \text{(C6)}$$

they are sufficient to derive the other constraints we need namely

$$ad + be + cf = 0 \qquad \text{(C7)}$$
$$gd + he + \text{if} = 0 \qquad \text{(C8)}$$

In the next two sections we will use these to decompose the matrix first assuming that there are no errors in t_{ij} terms and then with the realistic assumption that there are errors associated with the terms of the matrix T.

5.0 THE IDEAL CASE DECOMPOSITION

First we will give an outline of the technique and then the details.

5.1 SOLUTION OUTLINE

t_{31}, t_{32}, t_{33} will be used to solve for the three unknowns θ, ϕ and q. Once we know these, the six terms t_{11}, t_{12}, t_{13}, t_{21}, t_{22}, t_{23} can be expressed using only five unknowns k_1, k_2, u_0, v_0 and ψ. Further t_{11}, t_{12} and t_{13} can be expressed using only k_1, u_0 and ψ and similarly t_{21}, t_{22} and t_{23} can be expressed using only k_2, v_0 and ψ. Thus it is possible to compute k_1, u_0 and ψ from t_{11}, t_{12} and t_{13} and use the value of ψ and only two out of the three t_{21}, t_{22} and t_{23} to compute k_2 and v_0. The fact that we need only five out of these six terms is a consequence of having eleven t_{ij} but only ten independent variables. Finally t_{14} can be used to compute p and t_{24} can be used to compute r. Knowing p, q, r and θ, ϕ, ψ we can compute X_c, Y_c and Z_c.

5.2 SOLUTION DETAILS

Let us consider the sum

$$t_{31}^2 + t_{32}^2 + t_{33}^2$$

Since we have normalized all t_{ij} by dividing by t_{34} we have

$$t_{31} = \frac{d}{q}, \qquad t_{32} = \frac{e}{q}, \qquad t_{33} = \frac{f}{q}$$

Therefore, we get (using C2)

$$t_{31}^2 + t_{32}^2 + t_{33}^2 = \frac{1}{q^2}$$

Assuming q is positive (refer to Appendix A) we obtain q and hence d, e and f.

Let

$$\lambda_{31} = t_{31}q^2, \quad \lambda_{32} = t_{32}q^2, \quad \lambda_{33} = t_{33}q^2$$

and

$$\lambda_{11} = t_{11}\lambda_{32} - t_{12}\lambda_{31}$$

Hence

$$\lambda_{11} = \frac{k_1 a + u_0 d}{q}\,\frac{e}{q}\,q^2 - \frac{k_1 b + u_0 e}{q}\,\frac{d}{q}\,q^2$$
$$= k_1\,(ae - bd) = k_1 i$$

Similarly if we let

$$\lambda_{12} = t_{12}\lambda_{33} - t_{13}\lambda_{32}$$

and

$$\lambda_{13} = t_{13}\lambda_{31} - t_{11}\lambda_{33}$$

then it can be seen that

$$\lambda_{12} = k_1 g \quad \text{and} \quad \lambda_{13} = k_1 h$$

Squaring and adding we get (by using C3)

$$\lambda_{11}^2 + \lambda_{12}^2 + \lambda_{13}^2 = k_1^2\,(i^2 + g^2 + h^2) = k_1^2$$

Thus we obtain k_1^2 and hence k_1 (remember that k_1 is positive). Knowing k_1 we now obtain i, g and h from λ_{11}, λ_{12} and λ_{13}. Since we know i, g and h, using any two terms from t_{21}, t_{22} and t_{23} (say t_{21} and t_{22}) we can obtain k_2 and v_0 (see equations for t_{ij} in section 3). Note that one of the terms among t_{21}, t_{22} and t_{23} is redundant in the ideal case. That is to be expected considering that we only need ten equations to solve for the ten unknowns and we have been given eleven t_{ij}.

Using d, e and f (obtained from t_{31}, t_{32}, t_{33}) and the just obtained g, h and i we can compute a, b and c (using equations for a, b and c in terms of d, e, f, g, h and i in section 4). Using any of these three (say a) from t_{11} we obtain u_0. Using u_0 and k_1 in t_{14} we compute p . Similarly using v_0 and k_2 in t_{24} we can compute r .

Since $f = \sin\phi$ and $-\frac{\pi}{2} \le \phi \le +\frac{\pi}{2}$, we know ϕ. Since $e = \cos\theta \cos\phi$ and $d = -\sin\theta \cos\phi$, from d, e and ϕ we obtain $\cos\theta$ and $\sin\theta$ and hence θ. Similarly since $c = -\cos\phi \sin\psi$ and $i = \cos\phi \cos\psi$ from c, i and ϕ we obtain ψ.

So far we have computed k_1, k_2, u_0, v_0, θ, ϕ, ψ, p, q and r. The three X_c, Y_c and Z_c can be obtained using the following equations :

$$X_c = -ap - dq - gr$$

$$Y_c = -bp - eq - hr$$

$$Z_c = -cp - fq - ir$$

Thus it is possible to analytically compute the ten unknowns using t_{11}, t_{12}, t_{13}, t_{14}, t_{24}, t_{31}, t_{32}, t_{33} and any two from t_{21}, t_{22} and t_{23}. In much the same fashion we could have used t_{21}, t_{22}, t_{23}, t_{24}, t_{14}, t_{31}, t_{32}, t_{33} and any two from t_{11}, t_{12} and t_{13} to compute the same ten parameters by first computing k_2, then a, b and c, then k_1 and u_0, then v_0 and p and r . It should be emphasized here that we really need only ten of these eleven t_{ij} and if we are given all eleven t_{ij} of a transformation matrix the problem is harder to solve because the system is overconstrained. In the next section we describe an extension of this technique which obtains the ten parameters using all eleven t_{ij}.

6.0 DECOMPOSITION SOLUTION IN PRACTICE

In an actual case all terms t_{ij} have errors in them. One possible solution would be to discard any one of the six t_{11}, t_{12}, t_{13}, t_{21}, t_{22} and t_{23} and using the other five and t_{14}, t_{24}, t_{31}, t_{32} and t_{33} solve for the camera parameters. Thus we obtain six different solutions and either we have to average them or obtain a solution from these six by a least-square fit or averaging or some such technique. Not a very satisfying solution! Instead let us do the following. We have sixteen unknowns and eleven equations with six constraints on the terms a, b, c, d, e, f, g, h and i. From symmetry considerations there is no preference for dropping any one of t_{11}, t_{12}, t_{13}, t_{21}, t_{22}, t_{23} over the other five. Therefore let us not drop any! Instead we will relax the requirement that the terms a through i form an orthonormal matrix. This will be done by using five constraints instead of six by dropping one of the six constraints. But which one?

Let us examine the solution for the ideal case carefully. As before, we obtain q and hence d, e and f from t_{31}, t_{32} and t_{33}. This requires the use of the constraint

$$d^2 + e^2 + f^2 = 1$$

Define

$$\lambda_{11}, \quad \lambda_{12} \quad \text{and} \quad \lambda_{13}$$

as before and thus we obtain

$$k_1(ae-bd), \quad k_1(bf-ce) \quad \text{and} \quad k_1(cd-af)$$

Let us denote

$$(ae-bd) \text{ by } \bar{i}, \ (bf-ce) \text{ by } \bar{g} \text{ and } (cd-af) \text{ by } \bar{h}$$

Because of errors, these are not equal to i, g and h respectively but still

$$i \approx \bar{i}, \ g \approx \bar{g} \text{ and } h \approx \bar{h}$$

Let us use the following three constraints (one of these has already been used to obtain d, e and f)

$$ad+be+cf=0 \text{ and } a^2+b^2+c^2=1 \text{ and } d^2+e^2+f^2=1$$

These together imply (using the algebraic identity in section 4)

$$\bar{g}^2+\bar{h}^2+\bar{i}^2 = 1$$

and using these we obtain k_1 from

$$\lambda_{11}^2+\lambda_{12}^2+\lambda_{13}^2 = k_1^2 \ (\bar{g}^2+\bar{h}^2+\bar{i}^2) = k_1^2$$

and u_0 from

$$t_{11}^2+t_{12}^2+t_{13}^2 = \left(\frac{k_1}{q}\right)^2 (a^2+b^2+c^2) + \left(\frac{u_0}{q}\right)^2 (d^2+e^2+f^2) + 2\left(\frac{k_1}{q}\right)\left(\frac{u_0}{q}\right)(ad+be+cf) = \left(\frac{k_1}{q}\right)^2 + \left(\frac{u_0}{q}\right)^2$$

Thus we obtain k_1 (since k_1 is positive) and using that the magnitude of u_0 but not its sign. Similarly if we denote (as before)

$$\lambda_{21} = t_{21}\lambda_{32}-t_{22}\lambda_{31}, \quad \lambda_{22} = t_{22}\lambda_{33}-t_{23}\lambda_{32}$$

and

$$\lambda_{23} = t_{23}\lambda_{31}-t_{21}\lambda_{33}$$

we obtain

$$\lambda_{21} = k_2(ge-dh) = -k_2\bar{c} \ , \ \bar{c} \approx c$$

$$\lambda_{22} = k_2(fh-ei) = -k_2\bar{a} \ , \ \bar{a} \approx a$$

$$\lambda_{23} = k_2(id-fg) = -k_2\bar{b} \ , \ \bar{b} \approx b$$

We will now impose two more constraints in a similar fashion. They are

$$dg+eh+fi = 0 \quad \text{and} \quad g^2+h^2+i^2 = 1$$

These together imply

$$\bar{a}^2+\bar{b}^2+\bar{c}^2 = 1$$

and thus squaring and adding we get

$$\lambda_{21}^2+\lambda_{22}^2+\lambda_{23}^2 = k_2^2$$

and

$$t_{21}^2+t_{22}^2+t_{23}^2 = \left(\frac{k_2}{q}\right)^2 + \left(\frac{v_0}{q}\right)^2$$

Thus we can obtain the magnitudes of both k_2 and v_0 but not their signs. However using the approximate values $\bar{g}$ for g or $\bar{h}$ for h or $\bar{i}$ for i and substituting that in the equation for t_{21} or t_{22} or t_{23} (from section 3) we can determine the signs of k_2 and v_0. Similarly, using either the approximate value of $\bar{a}$ for a or $\bar{b}$ for b or $\bar{c}$ for c we can obtain the sign of u_0. (For this technique of determining signs to always work correctly, even in degenerate cases, care has to be exercised regarding the choice of the term for computation. One has to choose the term that has the maximum variation in magnitude when the sign of a variable in question is changed.)

Knowing the values of k_1, k_2, u_0 and v_0, we can use the expressions for t_{11}, t_{12}, t_{13}, t_{14}, t_{21}, t_{22}, t_{23} and t_{24} to calculate the values for a, b, c, p, g, h, i and r respectively. We now have the values for a, b, ..., i and also the values for p, q and r. Using equations (2.10a), (2.10b) and (2.10c) we can compute X_c, Y_c and Z_c. Using equations (2.4), (2.5) and (2.6) for d, e and f we can compute the angles θ and ϕ (as before). We can now use either the set of equations (2.1), (2.2) and (2.3) for a, b and c or the set of equations (2.7), (2.8) and (2.9) for g, h and i to compute the angle ψ. However, because of experimental errors, the values obtained in the two cases will in general be different. We could solve for both values and take their average to be the value for ψ. The difference between the two values is a measure of the consistency of the transformation matrix. Ideally it should be zero. It is possible to explain this deviation by assuming that the the two image axes U and V are not perpendicular to one another but that the two axes deviate from the perpendicular by an angle δ [21]. To summarize we have used all six t_{11}, t_{12}, t_{13}, t_{21}, t_{22} and t_{23} but used only the following five constraints

$$a^2+b^2+c^2 = d^2+e^2+f^2 = g^2+h^2+i^2 = 1$$

and

$$ad+be+cf = dg+eh+fi = 0$$

Note the symmetry in the choice of constraints. The constraint that we have not used is

$$ag+bh+ci = 0$$

In fact we will compute $ag+bh+ci$ to evaluate the consistency of the eleven given t_{ij}. It can be shown that the magnitude of this value should be the same as the magnitude of $\sin\delta$ [22]. The visualization of δ as the skew angle is a convenient way to characterize the errors in the system. The closer it is to zero the less the errors in t_{ij}. (See Appendix B for an example.)

7.0 CONCLUSIONS

We have arrived at a very simple non-iterative algorithm for decomposing any given transformation matrix into the various camera parameters that constitute the components of the matrix. We have implemented the above algorithm as a C program running under the UNIX environment. The program has been tested extensively and performs well when given real data as well as erroneous and degenerate data.

8.0 ACKNOWLEDGEMENTS

Several members of Bell Laboratories have contributed in many ways to make this research possible. I would like to thank Arun Netravali and Jakub Segen for some fruitful discussions. I am particularly indebted to Mike Oien and Frank Sinden of Bell Laboratories for a thorough and critical reading of the manuscript and several constructive suggestions. Finally I would like to thank John Jarvis for encouragement of this research and for some valuable suggestions during the course of this research.

9.0 REFERENCES

[1] L. G. Roberts, "Machine Perception Of Three-Dimensional Solids," MIT Lincoln Lab, Technical Report No 315 (May 1963).

[2] I. Sobel, "On Calibrating Computer Controlled Cameras for perceiving 3-D Scenes," *Artificial Intelligence* 5 (1974) pp. 185-198.

[3] R. M. Haralick, "Using Perspective Transformations in Scene Analysis," *Computer Graphics and Image Processing,* 13 (1980) pp. 191-221.

[4] R. O. Duda and P. E. Hart, *Pattern Recognition and Scene Analysis*, New York, Wiley, 1973.

[5] I. E. Sutherland, "Three-dimensional Data Input by Tablet," *Proceedings IEEE* 62(4), April 1974, pp. 453-461.

[6] D. F. Rogers and J. A. Adams, *Mathematical Elements for Computer Graphics*, McGraw-Hill, New York, 1976.

[7] D. H. Ballard and C. M. Brown, *Computer Vision*, Prentice-Hall, New Jersey, 1982.

[8] S. Ganapathy, "Reconstruction of Scenes Containing Polyhedra From Stereo Pair of Views," Stanford A.I. Lab. Memo AIM-272, December 1975.

[9] M. J. Hannah, "Bootstrap Stereo," *Proceedings Image Understanding Workshop*, April 1980, pp. 201-208.

[10] D. B. Gennery, "Stereo-Camera Calibration," *Proceedings Image Understanding Workshop*, November 1979, pp. 101-108.

[11] H. P. Moravec, *Robot Rover Visual Navigation*, UMI Research Press, Ann Arbor, Michigan 1983.

[12] J. E. Roach and J. K. Aggarwal, "Determining the Movement of Objects from a Sequence of Images,"*IEEE Transactions on PAMI*, November 1979.

[13] T. S. Huang and R. Y. Tsai, "Image Sequence Analysis: Motion Estimation," Chapter 1 *Image Sequence Analysis*, ed. T. S. Huang, Springer-Verlag 1981.

[14] E. Church, "Revised Geometry of the Aerial Photograph - Bulletin of Aerial Photogrammetry No. 15," Syracuse University, 1945.

[15] P. R. Wolf, *Elements of Photogrammetry*, McGraw-Hill, New York, 1974.

[16] M. A. Fischler and R. C. Bolles, "Random Sample Consensus: A Paradigm For Model Fitting Applications To Image Analysis and Automated Cartography," *Proceedings Image Understanding Workshop*, April 1980, pp. 71-88.

[17] S. Ganapathy, "Camera Location Determination Problem," Working Paper, Bell Labs, October 1983.

[18] D. G. Lowe, "Solving For The Parameters Of Object Models From Image Descriptions," *Proceedings Image Understanding Workshop*, April 1980, pp. 121-127.

[19] M. Potmesil, Private Communication.

[20] H. Goldstein, *Classical Mechanics*, Addison Wesley, Reading, Massachusetts, 1981.

[21] M. Oien, Personal Communication.

[22] S. Ganapathy, "Camera Calibration and Decomposition Problems," Working Paper, Bell Labs, July 1983.

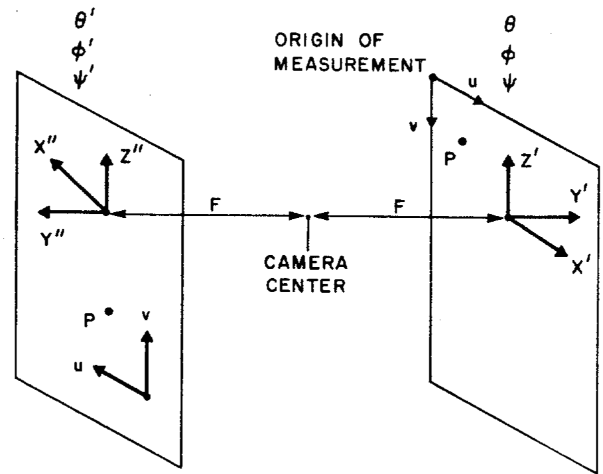


FIGURE 3 – THE TWO PLANES ARE ON THE OPPOSITE SIDES OF THE CAMERA CENTER. THE ANGLES ARE RELATED BY $\theta' = \theta + \pi$, $\phi' = -\phi$ AND $\psi' = -\psi$

APPENDIX A

We have tacitly assumed that q is always positive. Clearly when the camera is looking towards the origin q is positive. However, when the camera is looking away from the origin, q is negative. In such a case if we consider the image plane to be at a distance F from the camera but in the opposite direction (see figure 3), then q can be made positive. Mathematically this inversion can be accomplished as follows:

If q is negative, let $q' = -q$ so that q' is positive. Now if let $\lambda_{ij} = -t_{ij}$ then we get a new transformation matrix in which all terms have changed signs. In particular now $\lambda_{34} = -q = q'$ and is positive. The λ_{ij} can be written as

$$
\begin{aligned}
\lambda_{11} &= -k_1 a - u_0 d = k_1\,(-a) + u_0\,(-d)\\
\lambda_{12} &= -k_1 b - u_0 e = k_1\,(-b) + u_0\,(-e)\\
\lambda_{13} &= -k_1 c - u_0 f = k_1\,(-c) + u_0\,(-f)\\
\lambda_{14} &= -k_1 p - u_0 q = k_1\,(-p) + u_0\,q'\\
\lambda_{21} &= -k_2 g - v_0 d = -\,k_2\,(g) + v_0\,(-d)\\
\lambda_{22} &= -k_2 h - v_0 e = -\,k_2\,(h) + v_0\,(-e)\\
\lambda_{23} &= -k_2 i - v_0 f = -\,k_2\,(i) + v_0\,(-f)\\
\lambda_{24} &= -k_2 r - v_0 q = -\,k_2\,(r) + v_0\,q'\\
\lambda_{31} &= -d\\
\lambda_{32} &= -e\\
\lambda_{33} &= -f\\
\lambda_{34} &= q'
\end{aligned}
$$

In this new matrix q' is positive. Note carefully that this change was accomplished by changing the signs of a, b, c, d, e and f and k_2. The signs of g, h and i were not changed. This is necessary to assure that the rotation matrix is a "proper orthonormal" matrix. If we were to change the signs of all components, the determinant will also change sign and the matrix will no longer be "proper". This change in two rows of the matrix is equivalent to $\theta' = \theta + \pi$, $\phi' = -\phi$ $\psi' = -\psi$. That this is indeed so can be verified from figure 3. Thus it is valid to assume that q is always positive. In fact, without loss of generality, it is possible to assume that any two among the three k_1, k_2 and q are positive.

APPENDIX B

Given below is an experimentally obtained 3 by 4 transformation matrix

$$\begin{bmatrix} -2.3819E+00 & +4.9648E-01 & -3.9462E-02 & +8.4740E+02 \\ -4.3897E-02 & -6.2872E-02 & -2.4071E+00 & +8.8291E+02 \\ -2.6388E-04 & -6.2759E-04 & -7.1843E-05 & +1.0000E+00 \end{bmatrix}$$

Let us apply the algorithm in section 6 to the above example. From

$$t_{31} = -0.2638782E-03, \quad t_{32} = -0.6275908E-03$$

and

$$t_{33} = -0.7184327E-04$$

we get

$$q = (t_{31}^2 + t_{32}^2 + t_{33}^2)^{\frac{1}{2}} = +1.460728E+03$$
$$\lambda_{11} = +3.469129E+03 = k_1\,\bar{i}$$
$$\lambda_{12} = -1.289543E+02 = k_1\,\bar{g}$$
$$\lambda_{13} = -3.429078E+02 = k_1\,\bar{h}$$

Thus

$$\lambda_{11}^2 + \lambda_{12}^2 + \lambda_{13}^2 = +1.216907E+07 = k_1^2$$

Since k_1 is known to be positive we get

$$k_1 = \sqrt{(1.216907E+07)} = 3.488420E+03$$

and

$$t_{11}^2 + t_{12}^2 + t_{13}^2 = +5.921379E+00 = \left(\frac{k_1}{q}\right)^2 + \left(\frac{u_0}{q}\right)^2$$

Hence

$$u_0 = \pm\ 6.823031E+02$$

Computation of λ_{21}, λ_{22} and λ_{23} yields

$$\lambda_{21} = +2.338259E+01 = -k_2\,\bar{c}$$
$$\lambda_{22} = -3.213798E+03 = -k_2\,\bar{a}$$
$$\lambda_{23} = +1.348604E+01 = -k_2\,\bar{b}$$

Hence we get

$$\lambda_{21}^2 + \lambda_{22}^2 + \lambda_{23}^2 = +1.214778E+07 = k_2^2$$

and

$$t_{21}^2 + t_{22}^2 + t_{23}^2 = +5.800260E+00 = \left(\frac{k_2}{q}\right)^2 + \left(\frac{v_0}{q}\right)^2$$

Therefore

$$k_2 = \pm\ 3.485366E+03$$

and

$$v_0 = \pm\ 4.779105E+02$$

We also know that

$$d = qt_{31} = -3.854530E-01$$
$$e = qt_{32} = -9.167364E-01$$
$$f = qt_{33} = -1.049431E-01$$

Since $\bar{i}$ has the largest numerical value among $\bar{g}$, $\bar{h}$ and $\bar{i}$, using the expression

$$t_{23} = \frac{k_2 i + v_0 f}{q}$$

and using the approximate value $\bar{i}$ for i and using the just obtained value for f we determine the signs of k_2 to be negative and v_0 to be positive. Similarly since $\bar{a}$ has the largest numerical value among $\bar{a}$, $\bar{b}$ and $\bar{c}$ using the approximate value $\bar{a}$ for a and the expression for t_{11}, we determine the sign of u_0 to be positive. Thus we have determined k_1, k_2, u_0, v_0, d, e and f in both magnitude and sign. The signs of a, b, c, g, h and i are known but we have only approximate values for the magnitudes. Using the expressions for t_{14} and t_{24} we calculate p and r to be

$$p = +6.913188E+01$$
$$r = -1.697378E+02$$

Now that we know k_1, k_2, u_0 and v_0 exactly, the exact values of a, b, c, g, h and i can be determined from the values of t_{11}, t_{12}, t_{13}, t_{21}, t_{22} and t_{23}. These values can now be computed and they are shown below along with the approximate values and the difference (magnitude) between the two values

$a = -9.2198E-01$	$\bar{a} = -9.2208E-01$	$dif = 9.7212E-05$
$b = +3.8720E-01$	$\bar{b} = +3.8693E-01$	$dif = 2.6902E-04$
$c = +4.0016E-03$	$\bar{c} = +6.7087E-03$	$dif = 2.7071E-03$
$g = -3.4455E-02$	$\bar{g} = -3.6965E-02$	$dif = 2.5099E-03$
$h = -9.9352E-02$	$\bar{h} = -9.8298E-02$	$dif = 1.0536E-03$
$i = +9.9445E-01$	$\bar{i} = +9.9447E-01$	$dif = 1.4579E-05$

The values of d, e and f obviously are not affected. These nine values do not form an orthonormal matrix. (However, the nine values $\bar{a}$, $\bar{b}$, $\bar{c}$, d, e, f, $\bar{g}$, $\bar{h}$ and $\bar{i}$ do form a proper orthonormal matrix.) They do satisfy the five constraints

$$a^2 + b^2 + c^2 = d^2 + e^2 + f^2 = g^2 + h^2 + i^2 = 1$$

and

$$ad + be + cf = gd + he + if = 0$$

but the sixth constraint

$$ag + bh + ci$$

has a value of $-2.722193E-03$ instead of zero. This corresponds to a skew angle δ of approximately 0.156 (in degrees). This is a very small deviation and is consistent with the fact that the transformation matrix is quite accurate.

Knowing p, q and r we can now compute X_c, Y_c and Z_c to be

$$X_c = +6.209344E+02$$
$$Y_c = +1.295476E+03$$
$$Z_c = +3.218140E+02$$

From the values of a, b,... i we can compute the angles (in degrees) to be

$$\theta = +1.571951E+02$$
$$\phi = -6.023912E+00$$
$$\psi = +3.596915E+02$$

ψ is the average of the two values (see section 6). These have been verified to be close enough to the actual values and as a check we have given below the transformation matrix that would be obtained if we used the results of camera location, orientation, scaling etc to recompute the transformation matrix.

$$\begin{bmatrix} -2.3820E+00 & +4.9616E-01 & -3.6230E-02 & +8.4795E+02 \\ -4.0902E-02 & -6.4130E-02 & -2.4072E+00 & +8.8314E+02 \\ -2.6388E-04 & -6.2759E-04 & -7.1843E-05 & +1.0000E+00 \end{bmatrix}$$

A comparison of this matrix with the input matrix indicates reasonable agreement in values and confirms that the technique is computationally robust in the presence of errors in the terms of the transformation matrix.