

# Tsai's camera calibration method revisited

Berthold K.P. Horn

Copyright © 2000

## Introduction

Basic camera calibration is the recovery of the principle distance  $f$  and the principle point  $(x_0, y_0)^T$  in the image plane — or, equivalently, recovery of the position of the center of projection  $(x_0, y_0, f)^T$  in the image coordinate system. This is referred to as interior orientation in photogrammetry.

A calibration target can be imaged to provide correspondences between points in the image and points in space. It is, however, generally impractical to position the calibration target accurately with respect to the camera coordinate system using only mechanical means. As a result, the relationship between the target coordinate system and the camera coordinate system typically *also* needs to be recovered from the correspondences. This is referred to as exterior orientation in photogrammetry.

Since cameras often have appreciable geometric distortions, camera calibration is often taken to include the recovery of power series coefficients of these distortions. Furthermore, an unknown scale factor in image sampling may also need to be recovered, because scan lines are typically resampled in the frame grabber, and so picture cells do not correspond discrete sensing elements.

Note that in camera calibration we are trying to recover the transformations, based on measurements of coordinates, where one more often uses known transformation to map coordinates from one coordinate system to another.

Tsai's method for camera calibration recovers the interior orientation, the exterior orientation, the power series coefficients for distortion, and an image scale factor that best fit the measured image coordinates corresponding to known target point coordinates. This is done in stages, starting off with closed form least-squares estimates of some parameters and ending with an iterative non-linear optimization of all parameters simultaneously using these estimates as starting values. Importantly, it is error in the image plane that is minimized.

Details of the method are different for planar targets than for targets occupying some volume in space. Accurate planar targets are easier to make, but lead to some limitations in camera calibration, as pointed out below.

## Interior Orientation — Camera to Image

Interior Orientation is the relationship between camera-centric coordinates and image coordinates. The camera coordinate system has its origin at the center of projection, its  $z$  axis along the optical axis, and its  $x$  and  $y$  axes parallel to the  $x$  and  $y$  axes of the image.

Camera coordinates and image coordinates are related by the perspective projection equations:

$$\frac{x_I - x_0}{f} = \frac{x_C}{z_C} \quad \text{and} \quad \frac{y_I - y_0}{f} = \frac{y_C}{z_C}$$

where  $f$  is the principle distance (distance from the center of projection to the image plane), and  $(x_0, y_0)$  is the principle point (foot of the perpendicular from the center of projection to the image plane). That is, the center of projection is at  $(x_0, y_0, f)^T$ , as measured in the image coordinate system.

Interior orientation has three degrees of freedom. The problem of interior orientation is the recovery of  $x_0$ ,  $y_0$ , and  $f$ . This is the basic task of camera calibration. However, as indicated above, in practice we also need to recover the position and attitude of the calibration target in the camera coordinate system.

## Exterior Orientation — Scene to Camera

Exterior Orientation is the relationship between a scene-centered coordinate system and a camera-centered coordinate system. The transformation from scene to camera consists of a rotation and a translation. This transformation has six degrees of freedom, three for rotation and three for translation.

The scene coordinate system can be any system convenient for the particular design of the target. In the case of a planar target, the  $z$  axis is chosen perpendicular to the plane, and  $z = 0$  in the target plane.

If  $\mathbf{r}_S$  are the coordinates of a point measured in the scene coordinate system and  $\mathbf{r}_C$  coordinates measured in the camera coordinate system, then

$$\mathbf{r}_C = R(\mathbf{r}_S) + \mathbf{t}$$

where  $\mathbf{t}$  is the translation and  $R(\dots)$  the rotation.

If we chose for the moment to use an orthonormal matrix to represent rotation, then we can write this in component form:

$$\begin{pmatrix} x_C \\ y_C \\ z_C \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} x_S \\ y_S \\ z_S \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix}$$

where  $\mathbf{r}_C = (x_C, y_C, z_C)^T$ ,  $\mathbf{r}_S = (x_S, y_S, z_S)^T$ , and  $\mathbf{t} = (t_x, t_y, t_z)^T$ .

The unknowns to be recovered in the problem of exterior orientation are the translation vector  $\mathbf{t}$  and the rotation  $R(\dots)$ .

## The Unknown Horizontal Scale Factor

A complicating factor in the calibration of many modern electronic cameras is that the discrete nature of image sampling is not preserved in the signal.

In typical CCD or CMOS cameras, the initially discrete (staircase) sensor signal in analog form is low pass filtered to produce a smooth video output signal in standard form that hides the transitions between cells of the sensor. This waveform is then digitized in the frame grabber. The sampling in the horizontal direction in the frame grabber is typically *not* equal to the spacing of sensor cells, and is not known accurately. The horizontal spacing between pixels in the sampled image do not in general correspond to the horizontal spacing between cells in the image sensor.

This is in contrast with the vertical direction where sampling is controlled by the spacing of rows of sensor cells. Some digital cameras avoid the intermediate analog waveform and the low pass filtering, but many cameras — particularly cheaper ones intended for the consumer market — do not.

In this case the ratio of picture cell size in the horizontal and in the vertical direction is not known a priori from the dimensions of the sensor cells and needs to be determined. This can be done separately using frequency domain methods exploiting limitations of the approximate low pass filter and resulting aliasing effects.

Alternatively, the extra scaling parameter can be recovered as part of the camera calibration process. In this case we use a modified equation for  $x_I$ :

$$\frac{x_I - x_0}{f} = s \frac{x_C}{z_C}$$

where  $s$  is the unknown ratio of the pixel spacing in the  $x$ - and  $y$ -directions

It is not possible to recover this extra parameter when using planar targets, as discussed below, and so it has to be estimated separately in that case.

## Combining Interior and Exterior Orientation

If we combine the equations for interior and exterior orientation we obtain:

$$\frac{x_I - x_0}{f} = s \frac{r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x}{r_{31}x_S + r_{32}y_S + r_{33}z_S + t_z}$$

$$\frac{y_I - y_0}{f} = \frac{r_{21}x_S + r_{22}y_S + r_{23}z_S + t_y}{r_{31}x_S + r_{32}y_S + r_{33}z_S + t_z}$$

## Distortion

Projection in an ideal imaging system is governed by the pin-hole model. Real optical systems suffer from a number of inevitable geometric distortions. In optical systems made of spherical surfaces, with centers along the optical axis, a geometric distortion occurs in the radial direction. A point is imaged at a distance from the principle point that is larger (pin-cushion distortion) or smaller (barrel distortion) than predicted by the perspective projection equations; the displacement increasing with distance from the center. It is small for directions that are near parallel to the optical axis, growing as some power series of the angle. The distortion tends to be more noticeable with wide-angle lenses than with telephoto lenses.

The displacement due to radial distortion can be modelled using the equations:

$$\begin{aligned}\delta x &= x(\kappa_1 r^2 + \kappa_2 r^4 + \dots) \\ \delta y &= y(\kappa_1 r^2 + \kappa_2 r^4 + \dots)\end{aligned}$$

where  $x$  and  $y$  are measured from the center of distortion, which is typically assumed to be at the principle point. Only even powers of the distance  $r$  from the principle point occur, and typically only the first, or perhaps the first and the second term in the power series are retained.

Electro-optical systems typically have larger distortions than optical systems made of glass. They also suffer from tangential distortion, which is at right angle to the vector from the center of the image. Like radial distortion, tangential distortion grows with distance from the center of distortion.

$$\begin{aligned}\delta x &= -y(\epsilon_1 r^2 + \epsilon_2 r^4 + \dots) \\ \delta y &= +x(\epsilon_1 r^2 + \epsilon_2 r^4 + \dots)\end{aligned}$$

In calibration, one attempts to recover the coefficients  $(\kappa_1, \dots, \epsilon_1, \dots)$  of these power series. It is also possible to recover distortion parameters *separately* using, for example, the method of plumb lines.

Note that one can express the actual (distorted) image coordinates as a power series using predicted (undistorted) image coordinates as variables, or one can express predicted image coordinates as a power series in the actual image coordinates (that is, the  $r$  in the above power series can be either based on actual image coordinates or predicted image coordinates). The power series in the two cases are related by inversion.

If the power series to adjust distorted coordinates to undistorted coordinates is used, then it is more convenient to do the final optimization in undistorted image coordinates rather than distorted (actual) image coordinates.

If instead the power series to adjust undistorted coordinates to distorted coordinates is used, then it is more convenient to do the final optimization in distorted (actual) image coordinates rather than the undistorted image coordinates.

## Overall strategy

In calibration, a target of known geometry is imaged. Correspondences between target points and their images are obtained. These form the basic data on which the calibration is based.

Tsai's method first tries to obtain estimates of as many parameters as possible using linear least-squares fitting methods. This is convenient and fast since such problems can be solved using the pseudo-inverse matrix.

In this initial step, constraints between parameters (such as the orthonormality of a rotation matrix) are not enforced, and what is minimized is not the error in the image plane, but a quantity that simplifies the analysis and leads to linear equations. This does not affect the final result, however, since these estimated parameter values are used only as starting values for the final optimization.

In a subsequent step, the rest of the parameters are obtained using a non-linear optimization method that finds the best fit between the observed image points and those predicted from the target model. Parameters estimated in the first step are refined in the process.

Details of the calibration method are different when the target is planar than when it is not. Accurate planar targets are easier to make and maintain than three-dimensional targets, but limit calibration in ways that will become apparent. We start by analysing the case of the non-coplanar target.

## Estimating the rotation, and part of the translation

Initially we assume that we have a reasonable estimate of the position of the principle point  $(x_0, y_0)$ . This point is usually near the middle of the CCD or CMOS sensor. We refer coordinates to this point using

$$x'_I = x_I - x_0 \quad \text{and} \quad y'_I = y_I - y_0$$

so that

$$\frac{x'_I}{f} = s \frac{x_C}{z_C} \quad \text{and} \quad \frac{y'_I}{f} = \frac{y_C}{z_C}$$

Next, we consider only the *direction* of the point in the image as measured from the principle point. This yields a result that is independent of the unknown principle distance  $f$ . It is *also* independent of radial distortion.

$$\frac{x'_I}{y'_I} = s \frac{x_C}{y_C}$$

Using the expansion in terms of components of the rotation matrix  $R$  we obtain:

$$\frac{x'_I}{y'_I} = s \frac{r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x}{r_{21}x_S + r_{22}y_S + r_{23}z_S + t_y}$$

which becomes, after cross multiplying:

$$s(r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x)y'_I - (r_{21}x_S + r_{22}y_S + r_{23}z_S + t_y)x'_I = 0$$

or

$$(x_S y'_I)sr_{11} + (y_S y'_I)sr_{12} + (z_S y'_I)sr_{13} + y'_I st_x \\ - (x_S x'_I)r_{21} - (y_S x'_I)r_{22} - (z_S x'_I)r_{23} - x'_I t_y = 0$$

which we can view as a linear homogeneous equation in the eight unknowns

$$sr_{11}, sr_{12}, sr_{13}, r_{21}, r_{22}, r_{23}, st_x, \quad \text{and} \quad t_y$$

The coefficients in the equation are products of components of corresponding scene and image coordinates.

We obtain one such equation for every correspondence between a calibration target point  $(x_{S_i}, y_{S_i}, z_{S_i})^T$  and an image point  $(x_{I_i}, y_{I_i})^T$ .

Note that there is an unknown scale factor here because these equations are homogeneous. That is, if we have a solution for the eight unknowns, then any multiple of that solution is also a solution. In order to obtain a solution, we can convert the homogeneous equation into inhomogeneous equation by arbitrarily setting one unknown —  $t_y$  say — to one.

We then have seven unknowns for which we can solve if we have seven correspondences between target coordinates and image coordinates. If we have more than seven correspondences, we can minimize the sum of squares of errors using the pseudo-inverse method.

When we obtain the solution, we do have to remember that the eight unknowns (the seven we solved for plus the one we set to one) can be scaled by an arbitrary factor. Suppose that the best fit solution of the set of equations is

$$sr'_{11}, sr'_{12}, sr'_{13}, st'_x, r'_{21}, r'_{22}, r'_{23}, \quad \text{and} \quad t'_y = 1$$

We can estimate the correct scale factor by noting that the rows of the rotation matrix are supposed to be normal, that is

$$r_{11}^2 + r_{12}^2 + r_{13}^2 = 1 \quad \text{and} \quad r_{21}^2 + r_{22}^2 + r_{23}^2 = 1$$

We need to find the scale factor  $c$  to be applied to the solution to satisfy these equalities. It is easy to see that

$$c = 1/\sqrt{r'^2_{21} + r'^2_{22} + r'^2_{23}}$$

and

$$c/s = 1/\sqrt{(sr'_{11})^2 + (sr'_{12})^2 + (sr'_{13})^2}$$

These equations allow us to recover the factor  $c$  as well as the ratio  $s$  of the horizontal pixel spacing to the vertical pixel spacing.

Note that we did not enforce the orthogonality of the first two rows of the rotation matrix. We can improve matters by adjusting them to make them orthogonal.

### Forcing orthonormality of the first two rows

Given the vectors  $\mathbf{a}$  and  $\mathbf{b}$ , we can find two orthogonal vectors  $\mathbf{a}'$  and  $\mathbf{b}'$  that are as near as possible to  $\mathbf{a}$  and  $\mathbf{b}$  as follows:

$$\mathbf{a}' = \mathbf{a} + k\mathbf{b} \quad \text{and} \quad \mathbf{b}' = \mathbf{b} + k\mathbf{a}$$

$$\mathbf{a}' \cdot \mathbf{b}' = \mathbf{a} \cdot \mathbf{b} + k(\mathbf{a} \cdot \mathbf{a} + \mathbf{b} \cdot \mathbf{b}) + k^2\mathbf{a} \cdot \mathbf{b} = 0$$

The solution of this quadratic for  $k$  tends to be numerically ill behaved since the first and last coefficients ( $\mathbf{a} \cdot \mathbf{b}$ ) are small when the vectors are already close to being orthogonal. The following approximate solution can be used then

$$k \approx -(1/2)\mathbf{a} \cdot \mathbf{b}$$

(since  $k$  is expected to be small and  $\mathbf{a} \cdot \mathbf{a}$  and  $\mathbf{b} \cdot \mathbf{b}$  near one).

We have to renormalize the first two rows of the rotation matrix after adjusting them to make them orthogonal.

Finally we can obtain the third row of the rotation matrix simply by taking the cross-product of the first two rows. The resulting  $3 \times 3$  matrix will be orthonormal if we performed the above orthonormalization of the first two rows.

Note that there may be problems with this method of solution if the unknown ( $t_y$ ) that we decided to set equal to one in order to solve the homogeneous equations happens to be near zero. In this case a solution may be obtained by setting another unknown ( $t_x$  perhaps) equal to one. An alternative is to translate the experimental data by some offset in  $y$ .

Also note that we did not minimize the error in the image, but some other quantity that led to convenient, linear equations. The resulting rotation matrix and translation components are not especially accurate as a result. This is acceptable *only* because we use the recovered values merely as estimates in the full non-linear optimization described below.

### Coplanar Target

The above method cannot be used as is when the target is planar. It turns out that we cannot recover the scale factor  $s$  in this case, so we assume that image coordinates have already been adjusted to account for any differences in scaling in the  $x$  and  $y$  directions.

With a planar target we can always arrange the coordinate system such that  $z_S = 0$  for points on the target. This means the products with  $r_{13}$ ,  $r_{23}$ , and  $r_{33}$

drop out of the equations for the image coordinates and we obtain:

$$\frac{x'_I}{y'_I} = \frac{r_{11}x_S + r_{12}y_S + t_x}{r_{21}x_S + r_{22}y_S + t_y}$$

which becomes, after cross multiplying:

$$(r_{11}x_S + r_{12}y_S + t_x)y'_I - (r_{21}x_S + r_{22}y_S + t_y)x'_I = 0$$

or

$$(x_S y'_I) r_{11} + (y_S y'_I) r_{12} + y'_I t_x - (x_S x'_I) r_{21} - (y_S x'_I) r_{22} - x'_I t_y = 0$$

a linear homogeneous equation in the six unknowns

$$r_{11}, r_{12}, r_{21}, r_{22}, t_x, \text{ and } t_y$$

The coefficients in this equation are products of components of corresponding scene and image coordinates.

We obtain one such equation for every correspondence between a calibration target point  $(x_{S_i}, y_{S_i}, z_{S_i})^T$  and an image point  $(x_{I_i}, y_{I_i})^T$ .

As in the non-coplanar case, there is an unknown scale factor because these equations are homogeneous. If we have a solution for the six unknowns, then any multiple of that solution is also a solution. We can convert the homogeneous equations into inhomogeneous equations by arbitrarily setting one unknown —  $t_y$  say — to one.

We then have five unknowns for which we can solve if we have five correspondences between target coordinates and image coordinates. If we have more than five correspondences, we can minimize the sum of squares of errors using the pseudo-inverse.

We do have to remember though that the six unknowns (the five we solved for plus the one we set equal to one) can be scaled by an arbitrary factor.

Suppose that the best fit solution of the set of equations is

$$r'_{11}, r'_{12}, r'_{21}, r'_{22}, t'_x, \text{ and } t'_y = 1$$

We now are faced with the task of estimating the full  $3 \times 3$  rotation matrix based only on its top left  $2 \times 2$  sub-matrix.

## Recovering the full rotation matrix for planar target

We can estimate the correct scale factor by noting that the rotation matrix is supposed to be orthonormal and hence

$$r_{11}^2 + r_{12}^2 + r_{13}^2 = 1$$

$$r_{21}^2 + r_{22}^2 + r_{23}^2 = 1$$

$$r_{11}r_{21} + r_{12}r_{22} + r_{13}r_{23} = 0$$



so we have

$$r'_{11}{}^2 + r'_{12}{}^2 + r'_{13}{}^2 = k^2$$

$$r'_{21}{}^2 + r'_{22}{}^2 + r'_{23}{}^2 = k^2$$

$$r'_{11}r'_{21} + r'_{12}r'_{22} + r'_{13}r'_{23} = 0$$

From the first two equations we have

$$(r'_{11}{}^2 + r'_{12}{}^2)(r'_{21}{}^2 + r'_{22}{}^2) = (k^2 - r'_{13}{}^2)(k^2 - r'_{23}{}^2)$$

From the third equation we get

$$(r'_{11}r'_{21} + r'_{12}r'_{22})^2 = (r'_{13}r'_{23})^2$$

Subtracting we obtain

$$(r'_{11}r'_{22} - r'_{12}r'_{21})^2 = k^4 - k^2(r'_{13}{}^2 + r'_{23}{}^2)$$

Since

$$(r'_{13}{}^2 + r'_{23}{}^2) = 2k^2 - (r'_{11}{}^2 + r'_{12}{}^2 + r'_{21}{}^2 + r'_{22}{}^2)$$

we end up with

$$k^4 - k^2(r'_{11}{}^2 + r'_{12}{}^2 + r'_{21}{}^2 + r'_{22}{}^2) + (r'_{11}r'_{22} - r'_{12}r'_{21})^2 = 0$$

a quadratic in  $k^2$ . This then allows us to calculate the missing coefficients  $r'_{13}$  and  $r'_{23}$  in the first two rows of the rotation matrix using

$$r'_{13}{}^2 = k^2 - (r'_{11}{}^2 + r'_{12}{}^2)$$

$$r'_{23}{}^2 = k^2 - (r'_{21}{}^2 + r'_{22}{}^2)$$

Only the more positive of the two roots for  $k^2$  makes the right hand sides of these equations positive, so we only need to consider that root.

$$k^2 = \frac{1}{2} \left( (r'_{11}{}^2 + r'_{12}{}^2 + r'_{21}{}^2 + r'_{22}{}^2) + \sqrt{((r'_{11} - r'_{22})^2 + (r'_{12} + r'_{21})^2)((r'_{11} + r'_{22})^2 + (r'_{12} - r'_{21})^2)} \right)$$

We normalize the first two rows of the rotation matrix by dividing by  $k$ . Finally, we make up the third row by taking the cross-product of the first two rows.

There are, however, sign ambiguities in the calculation of  $r'_{13}$  and  $r'_{23}$ . We can get the sign of the product  $r'_{13}r'_{23}$  using the fact that

$$r_{13}r_{23} = -(r_{11}r_{21} + r_{12}r_{22})$$

so there is only a two-way ambiguity — but we do need to pick the proper sign to get the proper rotation matrix.

One way to pick the correct signs for  $r_{13}$  and  $r_{23}$  is to use the resulting transformation to project the target points back into the image. If the signs are correct, the predicted image positions will be close to the observed image positions. If the signs of these two components of the rotation matrix are picked incorrectly, then the first two components of the third row of the estimated rotation matrix will be wrong also, since that row is the cross-product of the first two. As a result

many predicted image points will lie in the wrong quadrant of the image. We can test for this condition by taking dot-products of vectors in the image plane — measured from the estimated principle point — of corresponding measured and predicted image positions. We try the other signs for  $r_{13}$  and  $r_{23}$  if

$$\sum_{i=1}^N (x_{Ii}x_{Pi} + y_{Ii}y_{Pi}) < 0.$$

## Estimating principle distance and distance to the scene

So far we have estimated the rotation matrix  $R$  and the first two components of the translation ( $t_x$  and  $t_y$ ). We do not yet have estimates for the third component ( $t_z$ ) of the translation, or the principle distance  $f$ . We can estimate these two parameters starting from:

$$\frac{x'_I}{f} = s \frac{r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x}{r_{31}x_S + r_{32}y_S + r_{33}z_S + t_z}$$

$$\frac{y'_I}{f} = \frac{r_{21}x_S + r_{22}y_S + r_{23}z_S + t_y}{r_{31}x_S + r_{32}y_S + r_{33}z_S + t_z}$$

Cross multiplying we find

$$s(r_{11}x_S + r_{12}y_S + r_{13}z_S + t_x)f - x'_I t_z = (r_{31}x_S + r_{32}y_S + r_{33}z_S)x'_I$$

$$(r_{21}x_S + r_{22}y_S + r_{23}z_S + t_y)f - y'_I t_z = (r_{31}x_S + r_{32}y_S + r_{33}z_S)y'_I$$

Given that we have estimates for  $\{r_{ij}\}$ , we can treat these as linear equations in the two unknowns  $f$  and  $t_z$ . We can solve these equations for  $f$  and  $t_z$  using one or more correspondence between target and image. If we use many correspondences we can solve the resulting over-determined system using least-squares methods.

If the horizontal scale factor  $s$  is not known accurately we may want to only use the equations for  $y'_I$ , rather than equations for both  $x'_I$  and  $y'_I$ .

At this point we have estimates of the rotation matrix  $R$ , the translation vector  $\mathbf{t} = (t_x, t_y, t_z)^T$ , as well as the principle distance  $f$ . We still need to find the principle point  $(x_0, y_0)$  and the coefficients of the distortion power series. We also need to refine the parameters estimated so far, since these estimates are *not* based on minimization of the image error, but some convenient linear equations.

Note that the target must span a range of depth values ( $z_C$ ) in order to recover  $f$  and  $t_z$ . If target points are all at the same depth, then their image positions depend only on the ratio  $f/t_z$ . Hence  $f$  and  $t_z$  cannot be determined separately. Accuracy improves with the depth range of target points. In the case of a planar target this means that the target normal *must* be turned away from the optical axis.

## Non-linear optimization

At this point we minimize the image errors, that is, the difference between the observed image positions  $(x_I, y_I)^T$  and the positions  $(x_P, y_P)^T$  predicted based on the known target coordinates  $(x_S, y_S, z_S)^T$ . The parameters of interior orientation, exterior orientation and distortion are adjusted to minimize

$$\sum_{i=1}^N (x_{Ii} - x_{Pi})^2 + \sum_{i=1}^N (y_{Ii} - y_{Pi})^2$$

This is best done using iterative numerical optimization such as a modified Levenberg-Marquardt method.

Non-linear optimization methods work best when they have full access to the components of the error. In the case here, it is best to treat  $(x_{Ii} - x_{Pi})$  and  $(y_{Ii} - y_{Pi})$  as separate error components rather than, for example, lumping them into a combined error term of the form

$$\sqrt{(x_{Ii} - x_{Pi})^2 + (y_{Ii} - y_{Pi})^2}$$

## Representation of rotation

To use non-linear optimization methods we need a non-redundant parameterization for rotation. Orthonormal matrices are redundant since they use nine numbers to represent just three degrees of freedom. Maintaining the six constraints of orthonormality in the minimization is very difficult.

Tsai's original method used Euler angles to represent rotations. An alternative is the Gibb's vector

$$\hat{\omega} \tan(\theta/2)$$

The direction of the Gibb's vector is the axis about which the rotation takes place,  $\hat{\omega}$ , while its magnitude is the tangent of half the angle of rotation,  $\theta$ . Like all non-redundant representations for rotation, the Gibb's vector has a singularity. It occurs when the rotation is through  $\pi$  radians.

An alternative is to use a redundant representation that has no singularities — and enforce the required non-linear constraint. The axis-and-angle representation of rotation can be related to the unit quaternion notation. The unit quaternion for the rotation is

$$\hat{q} = (\cos(\theta/2), \hat{\omega} \sin(\theta/2))$$

The needed constraint that this be a unit vector is easily incorporated in the non-linear optimization by adding an error term of the form

$$(\hat{q} \cdot \hat{q} - 1)^2$$

## Sensitivity of solution

Error in the calibration parameters is directly proportional to error in image measurements. The proportionality factor depends on the imaging geometry, and on the design of the target. Some parameters of the calibration are more sensitive to error than others, and some behave worse when the field of view is narrow or the depth range of the target limited.

It is difficult to investigate the sensitivities to noise analytically because of the non-linearity of the imaging model. However, the sensitivity issue can be studied easily using Monte Carlo simulation. Add random noise to the calibration image data and recompute the calibration parameters. Repeat and collect mean and covariance statistics.

## References and Acknowledgements

- Horn, B.K.P. (1986) *Robot Vision*, MIT Press, Cambridge, Massachusetts and McGraw-Hill, New York.
- Horn, B.K.P. (1987) "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., H. M. Hilden & S. Negahdaripour, (1988) "Closed Form Solution of Absolute Orientation using Orthonormal Matrices," *Journal of the Optical Society A*, Vol. 5, No. 7, pp. 1127–1135, July.
- Horn, B.K.P. (1990) "Relative Orientation," *International Journal of Computer Vision*, Vol. 4, No. 1, pp. 59–78, January.
- Horn, B.K.P. (1991) "Relative Orientation Revisited," *Journal of the Optical Society of America, A*, Vol. 8, pp. 1630–1638, October.
- Tsai, Roger Y. (1986) "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami Beach, FL, 1986, pp. 364–374.
- Tsai, Roger Y. (1987) "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses," *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 4, August 1987, pp. 323–344.

An implementation of Tsai's method dating to 1995 by Reg Willson of 3M in St. Paul, MN may be found on the web. Willson further acknowledges Piotr Jasiobedzki, Jim Vaughan, Ron Steriti, Torfi Thorhallsson, Frederic Devernay, Volker Rodehorst, and Jon Owen.

The public domain MINPACK “lmdif” package can be used for the non-linear optimization. This uses a modified Levenberg-Marquardt algorithm with a Jacobian calculated by a forward-difference approximation.