6

Camera Calibration

For the ancient Egyptians, exactitude was symbolized by a feather that served as a weight on scales used for the weighing of souls.

Italo Calvino, *Six Memos for the Next Millennium*

This chapter tackles the problem of camera calibration; that is, determining the value of the extrinsic and intrinsic parameters of the camera.

Chapter Overview

Section 6.1 defines and motivates the problem of camera calibration and its main issues.

Section 6.2 discusses a method based on simple geometric properties for estimating the camera parameters, given a number of correspondences between scene and image points.

Section 6.3 describes an alternative, simpler method, which recovers the projection matrix first, then computes the camera parameters as functions of the entries of the matrix.

What You Need to Know to Understand this Chapter

- Working knowledge of geometric camera models (Chapter 2).
- SVD and constrained least-squares (Appendix, section A.6).
- Familiarity with line extraction methods (Chapter 5).

6.1 Introduction

We learned in Chapters 4 and 5 how to identify and locate image features, and we are therefore fully equipped to deal with the important problem of *camera calibration*; that

is, estimating the values of the intrinsic and extrinsic parameters of the camera model, which was introduced in Chapter 2.

The key idea behind calibration is to *write the projection equations linking the known coordinates of a set of 3-D points and their projections, and solve for the camera parameters*. In order to get to know the coordinates of some 3-D points, camera calibration methods rely on one or more images of a *calibration pattern*: that is, a 3-D object of known geometry, possibly located in a known position in space and generating image features which can be located accurately. Figure 6.1 shows a typical calibration pattern, consisting of two planar grids of black squares on a white background. It is easy to know the 3-D position of the vertices of each square once the position of the two planes has been measured, and locate the vertices on the images, for instance as intersection of image lines, thanks to the high contrast and simple geometry of the pattern.

Problem Statement

Given one or more images of a calibration pattern, estimate

1. the intrinsic parameters,
2. the extrinsic parameters, or
3. both.

☞ The accuracy of calibration depends on the accuracy of the measurements of the calibration pattern; that is, its construction tolerances. To be on the safe side, the calibration pattern should be built with tolerances one or two order of magnitudes smaller than the desired accuracy of calibration. For example, if the desired accuracy of calibration is 0.1mm, the calibration pattern should be built with tolerances smaller than 0.01mm.

Although there are techniques inferring 3-D information about the scene from *uncalibrated* cameras, some of which will be described in the next two chapters, effective camera calibration procedures open up the possibility of using a wide range of existing algorithms for 3-D reconstruction and recognition, all relying on the knowledge of the camera parameters.

This chapter discusses two algorithms for camera calibration. The first method recovers directly the intrinsic and extrinsic camera parameters. The second method estimates the projection matrix first, *without* solving explicitly for the various parameters,[1] which are then computed as closed-form functions of the entries of the projection matrix. The choice of which method to adopt depends largely on which algorithms are to be applied next (Section 6.4).

---

[1] Recall that the projection matrix links world and image coordinates, and its entries are functions of the intrinsic and extrinsic parameters.

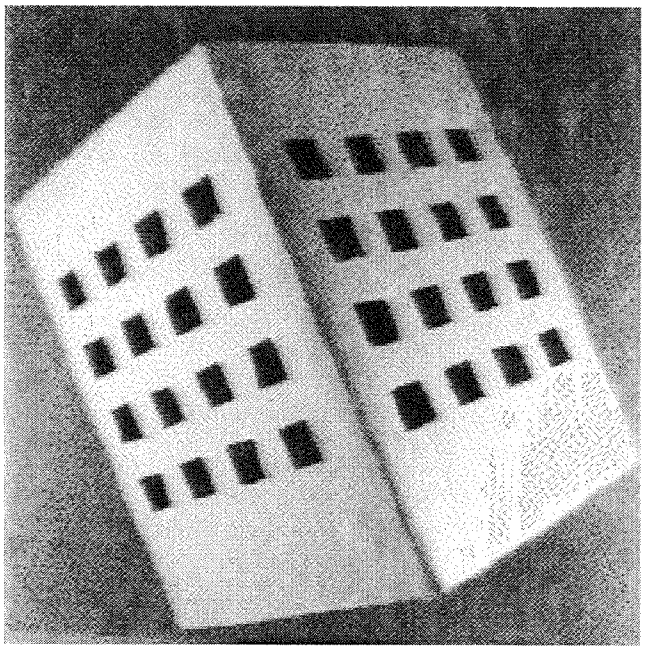

Figure 6.1 The typical calibration pattern used in this chapter.

6.2 Direct Parameter Calibration

We start by identifying the parameters to be estimated, and cast the problem in geometric terms.

6.2.1 Basic Equations

Consider a 3-D point, P, defined by its coordinates $[X^w, Y^w, Z^w]^\top$ in the world reference frame. As usual in calibration, the world reference frame is known.

☞ This means to pick an accessible object defining three mutually orthogonal directions intersecting in a common point. In this chapter, this object is the calibration pattern; in vision systems for indoor robot navigation, for instance, it can be a corner of a room.

Let $[X^c, Y^c, Z^c]^\top$ be the coordinates of P in the *camera* reference frame (with $Z^c > 0$ if P is visible). As usual, the origin of the camera frame is the center of projection, and the Z axis is the optical axis. The position and orientation of the camera frame is unknown, since, unlike the image and world reference frames, the camera frame is inaccessible directly. This is equivalent to saying that the we do not know the extrinsic parameters; that is, the 3×3 rotation matrix R and 3-D translation vector $\mathbf{T}$ such that

$$\begin{bmatrix} X^c \\ Y^c \\ Z^c \end{bmatrix} = R \begin{bmatrix} X^w \\ Y^w \\ Z^w \end{bmatrix} + \mathbf{T}. \tag{6.1}$$

In components, (6.1) can be written as

$$\begin{aligned} X^c &= r_{11}X^w + r_{12}Y^w + r_{13}Z^w + T_x \\ Y^c &= r_{21}X^w + r_{22}Y^w + r_{23}Z^w + T_y \\ Z^c &= r_{31}X^w + r_{32}Y^w + r_{33}Z^w + T_z. \end{aligned} \tag{6.2}$$

☞ Note the slight but important change of notation with respect to Chapter 2. In that chapter, the transformation between the world and camera reference frames was defined by translation *followed* by rotation. Here, the order is reversed and rotation *precedes* translation. While the rotation matrix is the same in both cases, the translation vectors differ (see Question 6.2).

Assuming that radial distortions (section 2.4.3) can be neglected,[2] we can write the image of $[X^c, Y^c, Z^c]^\top$ in the image reference frame as (see (2.14) and (2.20))

$$x_{im} = -\frac{f}{s_x}\frac{X^c}{Z^c} + o_x \tag{6.3}$$

$$y_{im} = -\frac{f}{s_y}\frac{Y^c}{Z^c} + o_y \tag{6.4}$$

For simplicity, and since there is no risk of confusion, we drop the subscript $_{im}$ indicating image (pixel) coordinates, and write (x, y) for (x_{im}, y_{im}). As we know from Chapter 2, (6.3) and (6.4) depend on the five intrinsic parameters f (focal length), s_x and s_y (horizontal and vertical effective pixel size), and o_x and o_y (coordinates of the image center), and, owing to the particular form of (6.3) and (6.4), the five parameters are not independent. However, if we let $f_x = f/s_x$ and $\alpha = s_y/s_x$, we may consider a new set of four intrinsic parameters, o_x, o_y, f_x, and α, all *independent* of one another. The parameter f_x is simply the focal length expressed in the effective horizontal pixel size (the *focal length in horizontal pixels*), while α, usually called *aspect ratio*, specifies the pixel deformation induced by the acquisition process defined in Chapter 2. Let us now summarize all the parameters to be calibrated in a box (see also the discussion in sections 2.4.2 and 2.4.3).

Extrinsic Parameters

- R, the 3×3 rotation matrix
- $\mathbf{T}$, the 3-D translation vector

---

[2] We shall reconsider this assumption in Exercise 6.1, which suggests how to calibrate the parameters of the radial distortion model of Chapter 2.

Intrinsic Parameters

- $f_x = f/s_x$, length in effective horizontal pixel size units
- $\alpha = s_y/s_x$, aspect ratio
- (o_x, o_y), image center coordinates
- k_1, radial distortion coefficient

Plugging (6.2) into (6.3) and (6.4) gives

$$x - o_x = f_x \frac{r_{11}X^w + r_{12}Y^w + r_{13}Z^w + T_x}{r_{31}X^w + r_{32}Y^w + r_{33}Z^w + T_z} \tag{6.5}$$

$$y - o_y = -f_y \frac{r_{21}X^w + r_{22}Y^w + r_{23}Z^w + T_y}{r_{31}X^w + r_{32}Y^w + r_{33}Z^w + T_z} \tag{6.6}$$

Notice that (6.5) and (6.6) bypass the inaccessible camera reference frame and link *directly* the world coordinates $[X^w, Y^w, Z^w]^\top$ with the coordinates (x, y) of the corresponding image point. If we use a known calibration pattern, both vectors are measurable. This suggests that, given a sufficient number of points on the calibration pattern, we can try to solve (6.5) and (6.6) for the unknown parameters. This is the idea behind the first calibration method, which is articulated in two parts:

1. assuming the coordinates of the image center are known, estimate all the remaining parameters
2. find the coordinates of the image center

6.2.2 Focal Length, Aspect Ratio, and Extrinsic Parameters

We assume that the coordinates of the image center are known. Thus, with no loss of generality, we can consider the translated coordinates $(x, y) = (x - o_x, y - o_y)$. In other words, we assume that the image center is the origin of the image reference frame. As we said in the introduction, the key idea is to exploit the known coordinates of a sufficient number of corresponding image and world points.

Assumptions and Problem Statement

Assuming that the location of the image center (o_x, o_y) is known, and that radial distortion can be neglected, estimate f_x, α, R, and $\mathbf{T}$ from image points (x_i, y_i), $i = 1 \ldots N$, projections of N known world points $[X_i^w, Y_i^w, Z_i^w]^\top$ in the world reference frame.

The key observation is that (6.5) and (6.6) have the same denominator; therefore, from each corresponding pair of points $((X_i^w, Y_i^w, Z_i^w),(x_i, y_i))$ we can write an equation of the form

$$x_i f_y (r_{21}X_i^w + r_{22}Y_i^w + r_{23}Z_i^w + T_y) = y_i f_x (r_{11}X_i^w + r_{12}Y_i^w + r_{13}Z_i^w + T_x). \tag{6.7}$$

Since $\alpha = f_x/f_y$, (6.7) can be thought of as a linear equation for the 8 unknowns $\mathbf{v} = (v_1, v_2, \ldots, v_8)$:

$$x_i X_i^w v_1 + x_i Y_i^w v_2 + x_i Z_i^w v_3 + x_i v_4 - y_i X_i^w v_5 - y_i Y_i^w v_6 - y_i Z_i^w v_7 - y_i v_8 = 0$$

where

$$\begin{aligned} v_1 &= r_{21} & v_5 &= \alpha r_{11} \\ v_2 &= r_{22} & v_6 &= \alpha r_{12} \\ v_3 &= r_{23} & v_7 &= \alpha r_{13} \\ v_4 &= T_y & v_8 &= \alpha T_x. \end{aligned}$$

Writing the last equation for the N corresponding pairs leads to the homogeneous system of N linear equations

$$A\mathbf{v} = 0 \tag{6.8}$$

where the $N \times 8$ matrix A is given by

$$A = \begin{bmatrix} x_1 X_1^w & x_1 Y_1^w & x_1 Z_1^w & x_1 & -y_1 X_1^w & -y_1 Y_1^w & -y_1 Z_1^w & -y_1 \\ x_2 X_2^w & x_2 Y_2^w & x_2 Z_2^w & x_2 & -y_2 X_2^w & -y_2 Y_2^w & -y_2 Z_2^w & -y_2 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ x_N X_N^w & x_N Y_N^w & x_N Z_N^w & x_N & -y_N X_N^w & -y_N Y_N^w & -y_N Z_N^w & -y_N \end{bmatrix}.$$

If $N \geq 7$ and the N points are not coplanar, A has rank 7, and system (6.8) has a nontrivial solution (unique up to an unknown scale factor), which can be determined from the SVD of A, $A = UDV^\top$, as the column of V corresponding to the only null singular value along the diagonal of D (Appendix, section A.6).

☞ The effects of the noise and the inaccurate localization of image and world points make the rank of A likely to be maximum (eight). In this case, the solution is the eigenvector corresponding to the *smallest* eigenvalue.

A rigorous proof of the fact that, in the ideal case (noise-free, perfectly known coordinates) the rank of A is 7 seems too involved to be presented here, and we refer you to the Further Readings section for details. Here, we just observe that, if the effective rank is larger than 7, system (6.8) would only have the trivial solution.

Our next task is to determine the unknown scale factor (and hence the various camera parameters) from the solution vector $\mathbf{v} = \bar{\mathbf{v}}$. If we call γ the scale factor, we have

$$\bar{\mathbf{v}} = \gamma(r_{21}, r_{22}, r_{23}, T_y, \alpha r_{11}, \alpha r_{12}, \alpha r_{13}, \alpha T_x). \tag{6.9}$$

Since $r_{21}^2 + r_{22}^2 + r_{23}^2 = 1$, from the first three components of $\bar{\mathbf{v}}$ we obtain

$$\sqrt{\bar{v}_1^2 + \bar{v}_2^2 + \bar{v}_3^2} = \sqrt{\gamma^2(r_{21}^2 + r_{22}^2 + r_{23}^2)} = |\gamma|. \tag{6.10}$$

Similarly, since $r_{11}^2 + r_{12}^2 + r_{13}^2 = 1$ and $\alpha > 0$, from the fifth, sixth, and seventh component of $\bar{\mathbf{v}}$ we have

$$\sqrt{\bar{v}_5^2 + \bar{v}_6^2 + \bar{v}_7^2} = \sqrt{\gamma^2\alpha^2(r_{11}^2 + r_{12}^2 + r_{13}^2)} = \alpha|\gamma|. \tag{6.11}$$

We can solve (6.10) and (6.11) for $|\gamma|$ as well as the aspect ratio α. We observe that the *first two rows* of the rotation matrix, R, and the first two components of the translation vector, $\mathbf{T}$, can now be determined, *up to an unknown common sign*, from (6.9). Furthermore, the *third row* of the matrix R can be obtained as the vector product of the first two estimated rows thought of as 3-D vectors. Interestingly, this implies that the *sign* of the third row is already fixed, as the entries of the third row remain unchanged if the signs of all the entries of the first two rows are reversed.

☞ Since the computation of the estimated rotation matrix, $\hat{R}$, does not take into account explicitly the orthogonality constraints, $\hat{R}$ cannot be expected to be orthogonal ($\hat{R}\hat{R}^\top = I$). In order to enforce orthogonality on $\hat{R}$, one can resort to the ubiquitous SVD decomposition. Assume the SVD of $\hat{R}$ is $\hat{R} = UDV^\top$. Since the three singular values of a 3×3 orthogonal matrix are all 1, we can simply replace D with the 3×3 identity matrix, I, so that the resulting matrix, $UIV^\top$, is exactly orthogonal (see Appendix, section A.6 for details).

Finally, we determine the unknown sign of the scale factor γ, and finalize the estimates of the parameters. To this purpose, we go back to (6.5), for example, with x instead of $x - o_x$, and recall that for every point $Z^c > 0$ and, therefore, x and $r_{11}X^w + r_{12}Y^w + r_{13}Z^w + T_x$ must have opposite sign. Consequently, it is sufficient to check the sign of $x(r_{11}X^w + r_{12}Y^w + r_{13}Z^w + T_x)$ for one of the points. If

$$x(r_{11}X^w + r_{12}Y^w + r_{13}Z^w + T_x) > 0, \tag{6.12}$$

the signs of the first two rows of $\hat{R}$ and of the first two components of the estimated translation vector must be reversed. Otherwise, no further action is required. A similar argument can be applied to y and $r_{21}X^w + r_{22}Y^w + r_{23}Z^w + T_y$ in (6.6).

At this point, we have determined the rotation matrix, R, the first two components of the translation vector, $\mathbf{T}$, and the aspect ratio, α. We are left with two, still undetermined parameters: T_z, the third component of the translation vector, and f_x, the focal length in horizontal pixel units. Both T_z and f_x can be obtained by least squares from a system of equations like (6.5) or (6.6), written for N points. To do this, for each point (x_i, y_i) we can write

$$x_i(r_{31}X_i^w + r_{32}Y_i^w + r_{33}Z_i^w + T_z) = -f_x(r_{11}X_i^w + r_{12}Y_i^w + r_{13}Z_i^w + T_x), \tag{6.13}$$

then solve the overconstrained system of N linear equations

$$A\begin{pmatrix} T_z \\ f_x \end{pmatrix} = \mathbf{b} \tag{6.14}$$

in the two unknowns T_z and f_x, where

$$A = \begin{bmatrix} x_1 & (r_{11}X_1^w + r_{12}Y_1^w + r_{13}Z_1^w + T_x) \\ x_2 & (r_{11}X_2^w + r_{12}Y_2^w + r_{13}Z_2^w + T_x) \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ x_N & (r_{11}X_N^w + r_{12}Y_N^w + r_{13}Z_N^w + T_x) \end{bmatrix}$$

and

$$b = \begin{pmatrix} -x_1(r_{31}X_1^w + r_{32}Y_1^w + r_{33}Z_1^w) \\ \cdot \\ \cdot \\ \cdot \\ -x_N(r_{31}X_N^w + r_{32}Y_N^w + r_{33}Z_N^w) \end{pmatrix}$$

The least squares solution $(\hat{T}_z, \hat{f}_x)$ of system (6.14) is

$$\begin{pmatrix} \hat{T}_z \\ \hat{f}_x \end{pmatrix} = (A^\top A)^{-1} A^\top \mathbf{b}. \tag{6.15}$$

It remains to be discussed how can we actually acquire an image of N points of known *world* coordinates, and locate the N corresponding image points accurately. One possible solution of this problem can be obtained with the pattern shown in Figure 6.1. The pattern consists of two orthogonal grids of equally spaced black squares drawn on white, perpendicular planes. We let the world reference frame be the 3-D reference frame centered in the lower left corner of the left grid and with the axes parallel to the three directions identified by the calibration pattern. If the horizontal and vertical size of the surfaces and the angle between the surfaces are known with high accuracy (from construction), then the 3-D coordinates of the vertices of each of the square in the *world* reference frame can be easily and accurately determined through simple trigonometry. Finally, the location of the vertices on the image plane can be found by intersecting the edge lines of the corresponding square sides. We now summarize the method:

explicit parameters calibration.

Algorithm EXPL_PARS_CAL

The input is an image of the calibration pattern described in the text (Figure 6.1) and the location of the image center.

1. Measure the 3-D coordinates of each vertex of the n squares on the calibration pattern in the world reference frame. Let $N = 4n$.
2. In order to find the coordinates in the *image* reference frame of each of the N vertices:
   - locate the image lines defined by the sides of the squares (e.g., using procedures EDGE_COMP and HOUGH_LINES of Chapters 3 and 4).
   - estimate the image coordinates of all the vertices of the imaged squares by intersecting the lines found.

3. Having established the N correspondences between image and world points, compute the SVD of A in (6.8). The solution is the column of V corresponding to the smallest singular value of A.
4. Determine $|\gamma|$ and α from (6.10) and (6.11).
5. Recover the first two rows of R and the first two components of $\mathbf{T}$ from (6.9).
6. Compute the third row of R as the vector product of the first two rows estimated in the previous step, and enforce the orthogonality constraint on the estimate of R through SVD decomposition.
7. Pick a point for which $(x - o_x)$ is noticeably different from 0. If inequality (6.12) is satisfied, reverse the sign of the first two rows of R and of the first two components of $\mathbf{T}$.
8. Set up A and $\mathbf{b}$ of system (6.14), and use (6.15) to estimate T_z and f_x.

The output is formed by α, f_x, and the extrinsic parameters of the viewing camera.

☞ When using a calibration pattern like the one in Figure 6.1, the 3-D squares lie on two different planes. The intersections of lines defined by squares from different world planes do not correspond to any image vertices. You must therefore ensure that your implementation considers only the intersections of pairs of lines associated to the *same* plane of the calibration pattern.

☞ The line equations in image coordinates are computed by least squares, using as many collinear edge points as possible. This process improves the accuracy of the estimates of line parameters and vertex location on the image plane.

6.2.3 Estimating the Image Center

In what follows we describe a simple procedure for the computation of the image center. As a preliminary step, we recall the definition of *vanishing points* from projective geometry, and state a simple theorem suggesting how to determine the image center through the orthocenter[3] of a triangle in the image.

Definition: Vanishing Points

Let $L_i, i = 1, \ldots, N$ be parallel lines in 3-D space, and l_i the corresponding image lines. Due to the perspective projection, the lines L_i appear to meet in a point p, called *vanishing point*, defined as the common intersection of all the image lines l_i.

Orthocenter Theorem: Image Center from Vanishing Points

Let T be the triangle on the image plane defined by the three vanishing points of three mutually orthogonal sets of parallel lines in space. The image center is the orthocenter of T.

The proof of the theorem is left as an exercise (Exercise 6.2). The important fact is, the theorem reduces the problem of locating the image center to one of intersecting

[3] The orthocenter of a triangle is the common intersection of the three altitudes.

image lines, and can be created easily on a suitable calibration pattern. In fact, we can use the same calibration pattern (actually, the same image!) of Figure 6.1, already used for EXPL_PARS_CAL, so that EXPL_PARS_CAL and the new algorithm, IMAGE_CENTER_CAL, fit nicely together.

Algorithm IMAGE_CENTER_CAL

The input is an image of the calibration pattern in Figure 6.1, and the output of the first two steps of algorithm EXPL_PARS_CAL.

1. Compute the three vanishing points p_1, p_2, and p_3, determined by the three bundles of lines obtained in step 2 of EXPL_PARS_CAL.
2. Compute the orthocenter, O, of the triangle $p_1p_2p_3$.

The output are the image coordinates of the image center, O.

☞ It is essential that the calibration pattern is imaged from a viewpoint guaranteeing that no vanishing point lies much farther than the others from the image center; otherwise, the image lines become nearly parallel, and small inaccuracies in the location of the lines result in large errors in the coordinates of the vanishing point. This can happen if one of the three mutually orthogonal directions is nearly parallel to the image plane, a situation to be definitely avoided. Even with a good viewpoint, it is best to determine the vanishing points using several lines and least squares.

☞ To improve the accuracy of the image center estimate, you should run IMAGE_CENTER_CAL with several views of the calibration patterns, and average the results.

Experience shows that an accurate location of the image center is not crucial for obtaining precise estimates of the other camera parameters (see Further Readings). Be careful, however, as accurate knowledge of the image center *is* required to determine the ray in space identified by an image point (as we shall see, for example, in Chapter 8).

6.3 Camera Parameters from the Projection Matrix

We now move on to the description of a second method for camera calibration. The new method consists in two sequential stages:

1. estimate the projection matrix linking world and image coordinates;
2. compute the camera parameters as closed-form functions of the entries of the projection matrix.

6.3.1 Estimation of the Projection Matrix

As we have seen in Chapter 2, the relation between the 3-D coordinates (X_i^w, Y_i^w, Z_i^w) of a point in space and the 2-D coordinates (x, y) of its projection on the image plane

can be written by means of a 3×4 projection matrix, M, according to the equation

$$\begin{pmatrix} u_i \\ v_i \\ w_i \end{pmatrix} = M \begin{pmatrix} X_i^w \\ Y_i^w \\ Z_i^w \\ 1 \end{pmatrix},$$

with

$$\begin{aligned} x &= \frac{u_i}{w_i} = \frac{m_{11}X_i^w + m_{12}Y_i^w + m_{13}Z_i^w + m_{14}}{m_{31}X_i^w + m_{32}Y_i^w + m_{33}Z_i^w + m_{34}} \\ y &= \frac{v_i}{w_i} = \frac{m_{21}X_i^w + m_{22}Y_i^w + m_{23}Z_i^w + m_{24}}{m_{31}X_i^w + m_{32}Y_i^w + m_{33}Z_i^w + m_{34}} \end{aligned} \tag{6.16}$$

The matrix M is defined up to an arbitrary scale factor and has therefore only 11 independent entries, which can be determined through a homogeneous linear system formed by writing (6.16) for at least 6 world-image point matches. However, through the use of calibration patterns like the one in Figure 6.1, many more correspondences and equations can be obtained and M can be estimated through least squares techniques. If we assume we are given N matches for the homogeneous linear system we have

$$A\mathbf{m} = 0, \tag{6.17}$$

with

$$A = \begin{bmatrix} X_1^w & Y_1^w & Z_1^w & 1 & 0 & 0 & 0 & 0 & -x_1X_1^w & -x_1Y_1^w & -x_1Z_1^w & -x_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1Z_1 & -y_1 \\ X_2^w & Y_2^w & Z_2 & 1 & 0 & 0 & 0 & 0 & -x_2X_2 & -x_2Y_2 & -x_2Z_2 & -x_2 \\ 0 & 0 & 0 & 0 & X_2 & Y_2 & Z_2 & 1 & -y_2X_2 & -y_2Y_2 & -y_2Z_2 & -y_2 \\ . & . & . & . & . & . & . & . & . & . & . & \\ . & . & . & . & . & . & . & . & . & . & . & \\ . & . & . & . & . & . & . & . & . & . & . & \\ X_N & Y_N & Z_N & 1 & 0 & 0 & 0 & 0 & -x_NX_N & -x_NY_N & -x_NZ_N & -x_N \\ 0 & 0 & 0 & 0 & X_N & Y_N & Z_N & 1 & -y_NX_N & -y_NY_N & -y_NZ_N & -y_N \end{bmatrix}$$

and

$$\mathbf{m} = [m_{11}, m_{12}, \ldots, m_{33}, m_{34}]^\top.$$

Since A has rank 11, the vector $\mathbf{m}$ can be recovered from SVD related techniques as the column of V corresponding to the zero (in practice the smallest) singular value of A, with $A = UDV^\top$ (see Appendix, section A.6). In agreement with the above definition of M, this means that the entries of M are obtained up to an unknown scale factor. The following is the detailed implementation of a method for estimating the matrix M:

Algorithm PROJ_MAT_CALIB

The input is an image of the calibration pattern described in the text (see Figure 6.1, for example).

1. Run the first two steps of EXPL_PARS_CAL.
2. Given N world-image matches, compute the SVD of A, system matrix of (6.17), $A = UDV^\top$. The solution $\mathbf{m}$ is the column of V corresponding to the smallest singular value of A.

The output is formed by the entries of the projection matrix, determined up to an unknown scale factor.

6.3.2 Computing Camera Parameters

We now want to express the intrinsic and extrinsic camera parameters as functions of the estimated projection matrix. To avoid confusion, we call $\hat{M}$ the projection matrix estimated through PROJ_MAT_CALIB (and hence $\hat{m}_{ij}$ the generic element of $\hat{M}$).

We first rewrite the full expression for the entries of M, or[4]

$$M = \begin{bmatrix} -f_x r_{11} + o_x r_{31} & -f_x r_{12} + o_x r_{32} & -f_x r_{13} + o_x r_{33} & -f_x T_x + o_x T_z \\ -f_y r_{21} + o_y r_{31} & -f_y r_{22} + o_y r_{32} & -f_y r_{23} + o_y r_{33} & -f_y T_y + o_y T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}. \quad (6.18)$$

☞ Notice that we are now using o_x, o_y, f_x, f_y as the four independent parameters, as opposed to $o_x, o_y, f_x, \alpha = s_y/s_x$ used in method 1.

☞ Notice also that not all 3×4 matrices can be written as functions of the extrinsic and intrinsic parameters as per 6.18. For details, see the Further Readings.

In what follows we also need the 3-D vectors

$$\mathbf{q}_1 = [\hat{m}_{11}, \hat{m}_{12}, \hat{m}_{13}]^\top,$$
$$\mathbf{q}_2 = [\hat{m}_{21}, \hat{m}_{22}, \hat{m}_{23}]^\top,$$
$$\mathbf{q}_3 = [\hat{m}_{31}, \hat{m}_{32}, \hat{m}_{33}]^\top,$$
$$\mathbf{q}_4 = [\hat{m}_{14}, \hat{m}_{24}, m_{34}]^\top.$$

Since M is defined up to a scale factor we can write

$$\hat{M} = \gamma M.$$

The absolute value of the scale factor, $|\gamma|$, can be obtained by observing that $\mathbf{q}_3$ is the last row of the rotation matrix, R. Hence,

$$\sqrt{\hat{m}_{31}^2 + \hat{m}_{32}^2 + \hat{m}_{33}^2} = |\gamma| \sqrt{r_{31}^2 + r_{32}^2 + r_{33}^2} = |\gamma|.$$

[4] These expressions were already written in Chapter 2 in order to link the perspective and weak-perspective camera models, but scaled by $1)f$, and under the simplifying assumptions $s_x = s_y = 1$, $o_x = o_y = 0$.

We now divide each entry of $\hat{M}$ by $|\gamma|$, and observe that the resulting, normalized projection matrix differs from M by, at most, a sign change. From now on, therefore, we indicate with $\hat{M}$ the normalized matrix. From the last row of (6.18) we have

$$T_z = \sigma \hat{m}_{34}$$

and

$$r_{3i} = \sigma \hat{m}_{3i}, \quad i = 1, 2, 3,$$

with $\sigma = \pm 1$. Taking the dot products of $\mathbf{q}_3$ with $\mathbf{q}_1$ and $\mathbf{q}_2$ we find

$$o_x = \mathbf{q}_1^\top \mathbf{q}_3$$

and

$$o_y = \mathbf{q}_2^\top \mathbf{q}_3.$$

Then we can recover f_x and f_y:

$$\begin{aligned} f_x &= \sqrt{\mathbf{q}_1^\top \mathbf{q}_1 - o_x^2} \\ f_y &= \sqrt{\mathbf{q}_2^\top \mathbf{q}_2 - o_y^2}, \end{aligned} \tag{6.19}$$

We can now compute the remaining extrinsic parameters as

$$\begin{aligned} r_{1i} &= \sigma(o_x \hat{m}_{3i} - \hat{m}_{1i})/f_x, \qquad i = 1, 2, 3 \\ r_{2i} &= \sigma(o_y \hat{m}_{3i} - \hat{m}_{2i})/f_y, \qquad i = 1, 2, 3 \\ T_x &= \sigma(o_x T_z - \hat{m}_{14})/f_x \\ T_y &= \sigma(o_y T_z - \hat{m}_{24})/f_y. \end{aligned} \tag{6.20}$$

As usual, the estimated rotation matrix, $\hat{R}$, is not really orthogonal, and we can find the closest orthogonal matrix as done in section 6.2.2.

☞ You may have noticed that M has 11 independent parameters, but there are only 6 extrinsic and 4 intrinsic parameters. The missing parameter is the angle, θ, formed by the axes of the image reference frame. Here we have exploited the fact that this angle is 90° within great accuracy in all commercial cameras, and is therefore not considered explicitly in the intrinsic parameters set. For details of both linear and nonlinear calibration methods estimating θ as well, see the Further Readings.

We are left to discuss how to determine the sign σ. The sign σ can be obtained from $T_z = \sigma \hat{m}_{34}$, because we know whether the origin of the world reference frame is in front of ($T_z > 0$) or behind ($T_z < 0$) the camera. We do not give the usual algorithm, which would merely restate the equations derived in this section.

6.4 Concluding Remarks

Is there any difference between the two calibration methods presented? Obviously, you should expect the same (in practice, very similar) parameter values from both! Method 2 is probably simpler, but the algorithm on which we based EXPL_PARS_CAL (see Further Readings) is a well-known technique in the computer vision community, and has been implemented and used by many groups. PROJ_MAT_CALIB is useful whenever the projection matrix is sufficient to solve a vision problem, and there is no need to make the individual parameters explicit; an example is referenced in the Further Readings.

We said that the precision of calibration depends on how accurately the world and image reference points are located. But *which* accuracy should one pursue? As the errors on the parameter estimates propagate to the results of the application, the answer depends ultimately on the accuracy requirements of the target application. For instance, inspection systems in manufacturing often require submillimeter accuracies, whereas errors of centimeters can be acceptable in vision systems for outdoors navigation. Locating image reference points or lines at pixel precision is generally unsatisfactory in the former case, but acceptable in the latter. A practical guideline is: *the effort going into improving calibration accuracy should be commensurate to the requirements of the application.*

6.5 Summary

After working through this chapter you should be able to:

- explain what calibration is and why it is necessary
- calibrate the intrinsic and extrinsic parameters of an intensity camera
- estimate the entries of the projection matrix
- design a calibration pattern, motivating your design choices

6.6 Further Readings

EXPL_PARS_CAL has been adapted from a well-known (but rather involved) paper by Tsai [10], which contains a proof that the rank of the matrix A of EXPL_PARS_CAL is 7 in the ideal case. The orthocenter property and the calibration of the image center through vanishing points is due to Caprile and Torre [2], who also suggest a neat method for calibrating the rotation matrix. Direct calibration of the projection matrix is described, for example, by Faugeras and Toscani [5]. The explicit computation of the calibration parameters from the projection matrix is taken from Faugeras's book [4], which also discusses the conditions under which this is possible (Chapter 3, Section 4). Ayache and Lustman [1] describe a stereo system which requires calibration of the projection matrix, but not explicit knowledge of the camera parameters. The influence on vision algorithms of erroneous estimates of the image center and other camera parameters is discussed by [3, 7, 6]. Thacker and Mayhew [9] describe a method for calibrating a stereo system from arbitrary stereo images.

Recent developments on calibration are presented, for instance, by Faugeras and Maybank [8] who introduce an elegant method based solely on point matches and able to obtain camera calibration without the use of calibration patterns.

6.7 Review

Questions

- ❒ *6.1* Describe the purpose of calibration and name applications in which calibration is necessary.
- ❒ *6.2* What is the relation between the translation vectors between two reference frames if the transformation between the frames is (a) first rotation and then translation and (b) *vice versa*? Verify your answer in the trivial case in which rotation is described by the identity matrix.
- ❒ *6.3* Why does algorithm EXPL_PARS_CAL determine only f_x and not f_y?
- ❒ *6.4* Discuss the construction requirements of a calibration pattern.
- ❒ *6.5* How would you estimate the accuracy of a calibration program?

Exercises

- ○ *6.1* In the case of large fields of view, or if pixels very far from the center of the image are considered, the radial distortion cannot be neglected. A possible procedure for calibrating the distortion parameter k_1 is to rewrite (6.13) with $x_i(1 + k_1 r_i^2)$ in place of x_i with $r_i^2 = x_i^2 + \alpha^2 y_i^2$, or

$$x_i(1 + k_1 r_i^2)(r_{31} X_i^w + r_{32} Y_i^w + r_{33} Z_i^w + T_z) = -f_x(r_{11} X_i^w + r_{12} Y_i^w + r_{13} Z_i^w + T_x).$$

  The corresponding nonlinear system of equations for the unknowns f_x, T_z, and k_1 can be solved through gradient descent techniques using the output of EXPL_PARS_CAL as initial guess for f_x and T_z and 0 as initial guess for k_1. Write out the complete equations suggested, and devise a calibration algorithm including k_1 in the intrinsic parameters.

- ○ *6.2* Prove the orthocenter theorem by geometrical arguments. (*Hint*[5]: Let h be the altitude from the vertex (and vanishing point) v to the side s, and O the projection center. Since both the segments h and vO are orthogonal to s, the plane through h and vO is orthogonal to s and hence to the image plane . . .)
- ○ *6.3* Prove that the vanishing points associated to three coplanar bundles of parallel lines are collinear.
- ○ *6.4* Estimate the theoretical error of the coordinates of the principal point as a function of the error (uncertainty) of the coordinates of the vanishing points. Can you guess the viewpoint that minimizes error propagation?

[5] Hint by Francesco Robbiano.