CHAPTER

Image transformations and camera calibration

19

When setting up a measurement system it is natural to calibrate it carefully before use. This task has been left to last because (1) it is mathematically more demanding, (2) there are instances where it can be bypassed, (3) it is not always possible to perform the calibration entirely in advance, but rather it has to be updated to a sufficient extent as measurements proceed. This chapter outlines some of the problems of calibration and some of the results of recent research which allow the process to be at least partially bypassed.

Look out for:

- the homogeneous coordinates technique for representing general 3-D positions and transformations
- "extrinsic" (external world) and "intrinsic" (camera) parameters
- methods for achieving absolute camera calibration
- the need for correction of camera lens distortions
- the idea of a generalized epipolar geometry
- the "essential" and "fundamental" matrix formulations, relating the observed positions of any point in two camera frames of reference
- the central position of the eight-point algorithm
- the possibility of image "rectification"
- the possibility of 3-D reconstruction.

This is one of the key chapters constituting Part 3 of this book. These chapters should be taken together as they involve not merely different topics but also different *aspects* of the subject, and in addition, the aim has been to cover them in as gentle an order as possible considering the mathematical complexities involved in extracting 3-D and motion information from 2-D images.

19.1 INTRODUCTION

When images are obtained from 3-D scenes, the exact position and orientation of the camera-sensing device is often unknown, and there is a need for it to be related to some global frame of reference. This is especially important if accurate measurements of objects are to be made from their images, e.g., in inspection

Computer Vision. DOI: http://dx.doi.org/10.1016/B978-0-12-809284-2.00019-8

applications. On the other hand, it may in certain cases be possible to dispense with such detailed information—as in the case of a security system for detecting intruders, or a system for counting cars on a motorway. There are also more complicated cases, such as those in which cameras can be rotated or moved on a robot arm, or the objects being examined can move freely in space. In such cases, camera calibration becomes a central issue. Before we can consider camera calibration, we need to understand in some detail the transformations that can occur between the original world points and the formation of the final image. We attend to these image transformations in the following section, and then move on to details of camera parameters and camera calibration in the subsequent two sections. Then, in Section 19.5, we consider how any radial distortions of the image introduced by the camera lens can be corrected.

Section 19.6 signals a break with previous work and introduces "multiple view" vision. This topic has become important in recent years, as it uses new theory to bypass the need for formal camera calibration, and makes it possible to update the vision system parameters during actual use. The basis for this work is generalized epipolar geometry: this takes the epipolar line ideas of Section 16.3.2 considerably further. At the core of this, new work are the "essential" and "fundamental" matrix formulations, which relate the observed positions of any point in two camera frames of reference. Short sections on image "rectification" (obtaining a new image as it would be seen from an idealized camera position) and 3-D reconstruction follow.

19.2 IMAGE TRANSFORMATIONS

First, we consider the rotations and translations of object points relative to a global frame. After a rotation through an angle θ about the Z-axis (Fig. 19.1), the coordinates of a general point (X, Y) change to:

$$X' = X\cos\theta - Y\sin\theta \tag{19.1}$$

$$Y' = X\sin\theta + Y\cos\theta \tag{19.2}$$

This result is neatly expressed by the matrix equation:

$$\begin{bmatrix} X' \\ Y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} \tag{19.3}$$

Clearly, similar rotations are possible about the X and Y axes. To satisfactorily express rotations in 3-D, we require a more general notation using 3×3 matrices, the matrix for a rotation θ about the Z-axis being:

$$\mathbf{Z}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{19.4}$$

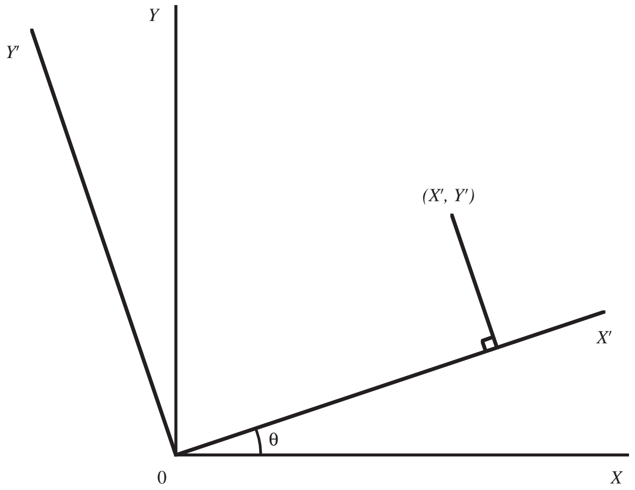

FIGURE 19.1

Effect of a rotation θ about the origin.

Those for rotations ψ about the X-axis and φ about the Y-axis are:

$$\mathbf{X}(\psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} \tag{19.5}$$

$$\mathbf{Y}(\varphi) = \begin{bmatrix} \cos\varphi & 0 & \sin\varphi \\ 0 & 1 & 0 \\ -\sin\varphi & 0 & \cos\varphi \end{bmatrix} \tag{19.6}$$

We can make up arbitrary rotations in 3-D by applying sequences of such rotations. Similarly, we can express arbitrary rotations as sequences of rotations about the coordinate axes. Thus, $\mathbf{R} = \mathbf{X}(\psi)\mathbf{Y}(\varphi)\mathbf{Z}(\theta)$ is a composite rotation in which $\mathbf{Z}(\theta)$ is applied first, then $\mathbf{Y}(\varphi)$, and finally $\mathbf{X}(\psi)$. Rather than multiplying out these matrices, we write down here the general result expressing an arbitrary rotation $\mathbf{R}$:

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \tag{19.7}$$

Note that the rotation matrix $\mathbf{R}$ is not completely general: it is orthogonal and thus has the property that $\mathbf{R}^{-1} = \mathbf{R}^{\mathrm{T}}$.

In contrast with rotation, translation through a distance (T_1, T_2, T_3) is given by:

$$X' = X + T_1 \tag{19.8}$$

$$Y' = Y + T_2 \tag{19.9}$$

$$Z' = Z + T_3 \tag{19.10}$$

which is not expressible in terms of a multiplicative 3 × 3 matrix. However, just as general rotations can be expressed as rotations about various coordinate axes, so general translations and rotations can be expressed as sequences of basic rotations and translations relative to individual coordinate axes. Thus, it would be most useful to have a notation which unified the mathematical treatment so that a generalized displacement could be expressed as a product of matrices. This is indeed possible if so-called *homogeneous coordinates* are used. To achieve this, the matrices must be augmented to 4 × 4. A general rotation can then be expressed in the form:

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & 0 \\ R_{21} & R_{22} & R_{23} & 0 \\ R_{31} & R_{32} & R_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{19.11}$$

whereas the general translation matrix becomes:

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T_1 \\ 0 & 1 & 0 & T_2 \\ 0 & 0 & 1 & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{19.12}$$

The generalized displacement (i.e., translation plus rotation) transformation clearly takes the form:

$$\begin{bmatrix} X' \\ Y' \\ Z' \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{19.13}$$

We now have a convenient notation for expressing generalized transformations including operations other than the translations and rotations which account for the normal motions of rigid bodies. First, we take a scaling in size of an object, the simplest case being given by the matrix:

$$\begin{bmatrix} S & 0 & 0 & 0 \\ 0 & S & 0 & 0 \\ 0 & 0 & S & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The more general case:

$$\begin{bmatrix} S_1 & 0 & 0 & 0 \\ 0 & S_2 & 0 & 0 \\ 0 & 0 & S_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

introduces a shear in which an object line λ will be transformed into a line that is not in general parallel to λ. Skewing is another interesting transformation, being given by linear translations varying from the simple case:

$$\begin{bmatrix} 1 & B & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

to the general case:

$$\begin{bmatrix} 1 & B & C & 0 \\ D & 1 & F & 0 \\ G & H & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Rotations can be regarded as combinations of scaling and skewing and are sometimes implemented as such (Weiman, 1976).

The other simple but interesting case is that of reflection, which is typified by:

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

This generalizes to other cases of improper rotation where the determinant of the top left 3×3 matrix is -1.

In all the cases discussed above, it will be observed that the bottom row of the generalized displacement matrix is redundant. In fact, we can put this row to good use in certain other types of transformation. Of particular interest in this context is the case of perspective projection. Following Section 16.3, Eq. (16.1), the equations for projection of object points into image points are as follows:

$$x = fX/Z \tag{19.14}$$

$$y = fY/Z \tag{19.15}$$

$$z = f \tag{19.16}$$

We next make full use of the bottom row of the transformation matrix by defining the homogeneous coordinates as $(X_{\mathrm{h}}, Y_{\mathrm{h}}, Z_{\mathrm{h}}, h) = (hX, hY, hZ, h)$, where h is a nonzero constant which we can take to be unity. To proceed, we examine the homogeneous transformation:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ Z/f \end{bmatrix} \tag{19.17}$$

We see that dividing by the fourth coordinate gives the required values of the transformed Cartesian coordinates $(fX/Z, fY/Z, f)$.

Let us now review this result. First, we have found a 4 × 4 matrix transformation which operates on 4-D homogeneous coordinates. These do not correspond directly to real coordinates, but real 3-D coordinates can be calculated from them by dividing the first three by the fourth homogeneous coordinate. Thus, there is an arbitrariness in the homogeneous coordinates in that they can all be multiplied by the same constant factor without producing any change in the final interpretation. Likewise, when deriving homogeneous coordinates from real 3-D coordinates, we can employ any convenient constant multiplicative factor h, though we will normally take h to be unity.

The advantage to be gained from use of homogeneous coordinates is the convenience of having a single multiplicative matrix for any transformation, in spite of the fact that perspective transformations are intrinsically nonlinear: thus, a quite complex nonlinear transformation can be reduced to a more straightforward linear transformation. This eases computer calculation of object coordinate transformations, and other computations such as those for camera calibration (see below). We may also note that almost every transformation can be inverted by inverting the corresponding homogeneous transformation matrix. The exception is the perspective transformation, for which the fixed value of z leads merely to Z being unknown, and X, Y only being known relative to the value of Z (hence the need for binocular vision or other means of discerning depth in a scene).

19.3 CAMERA CALIBRATION

The above discussion has shown how homogeneous coordinate systems are used to help provide a convenient linear 4 × 4 matrix representation for 3-D transformations including rigid body translations and rotations, and nonrigid operations including scaling, skewing, and perspective projection. In this last case, it was implicitly assumed that the camera and world coordinate systems are identical, as the image coordinates were expressed in the same frame of reference. However, in general, the objects viewed by the camera will have positions which may be known in world coordinates, but which will not a priori be known in camera coordinates, as the camera will in general be mounted in a somewhat arbitrary position and will point in a somewhat arbitrary direction. Indeed, it may well be on adjustable gimbals, and may also be motor driven, with no precise calibration system. If the camera is on a robot arm, there are likely to be position sensors which could inform the control system of the camera position and orientation in world coordinates, though the amount of slack may well make the information too imprecise for practical purposes (e.g., to guide the robot toward objects).

These factors mean that the camera system will have to be calibrated very carefully before the images can be used for practical applications such as robot pick-and-place. A useful approach is to assume a general transformation between the world coordinates and the image seen by the camera under perspective projection, and to locate in the image various calibration points which have been placed in known positions in the scene. If enough such points are available, it should be possible to compute the transformation parameters, and then all image points can be interpreted accurately until recalibration becomes necessary.

The general transformation **G** takes the form:

$$\begin{bmatrix} X_\mathrm{H} \\ Y_\mathrm{H} \\ Z_\mathrm{H} \\ H \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} & G_{13} & G_{14} \\ G_{21} & G_{22} & G_{23} & G_{24} \\ G_{31} & G_{32} & G_{33} & G_{34} \\ G_{41} & G_{42} & G_{43} & G_{44} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{19.18}$$

where the final Cartesian coordinates appearing in the image are $(x, y, z) = (x, y, f)$, and these are calculated from the first three homogeneous coordinates by dividing by the fourth:

$$x = X_\mathrm{H}/H = (G_{11}X + G_{12}Y + G_{13}Z + G_{14})/(G_{41}X + G_{42}Y + G_{43}Z + G_{44}) \tag{19.19}$$

$$y = Y_\mathrm{H}/H = (G_{21}X + G_{22}Y + G_{23}Z + G_{24})/(G_{41}X + G_{42}Y + G_{43}Z + G_{44}) \tag{19.20}$$

$$z = Z_\mathrm{H}/H = (G_{31}X + G_{32}Y + G_{33}Z + G_{34})/(G_{41}X + G_{42}Y + G_{43}Z + G_{44}) \tag{19.21}$$

However, as we know z, there is no point in determining parameters G_{31}, G_{32}, G_{33}, G_{34}. Accordingly, we proceed to develop the means for finding the other parameters. In fact, because only the ratios of the homogeneous coordinates are meaningful, only the ratios of the G_{ij} values need be computed, and it is usual to take G_{44} as unity: this leaves only 11 parameters to be determined. Multiplying out the first two equations and rearranging gives:

$$G_{11}X + G_{12}Y + G_{13}Z + G_{14} - x(G_{41}X + G_{42}Y + G_{43}Z) = x \tag{19.22}$$

$$G_{21}X + G_{22}Y + G_{23}Z + G_{24} - y(G_{41}X + G_{42}Y + G_{43}Z) = y \tag{19.23}$$

Noting that a single world point (X, Y, Z) which is known to correspond to image point (x, y) gives us *two* equations of the above form; it requires a minimum of six such points to provide values for all 11 G_{ij} parameters: Fig. 19.2 shows a convenient near-minimum case. An important factor is that the world points used for the calculation should lead to independent equations: thus, it is important that they should not be coplanar. More precisely, there must be at least six points, no four of which are coplanar. However, further points are useful in that they lead to overdetermination of the parameters and increase the accuracy with which the latter can be computed. There is no reason why the additional points should not be coplanar with existing points: indeed, a common arrangement is to set up a cube so that three of its faces are visible, each face having a pattern of squares with 30–40 easily discerned corner features (as for a Rubic cube).

Least squares analysis can be used to perform the computation of the 11 parameters, e.g., via the pseudoinverse method. First, the $2n$ equations have to be expressed in matrix form:

$$\mathbf{Ag} = \boldsymbol{\xi} \tag{19.24}$$

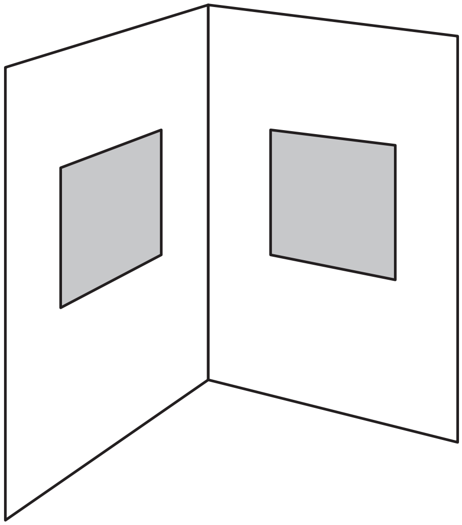

FIGURE 19.2

A convenient near-minimum case for camera calibration. Here, two sets of four coplanar points, each set of four being at the corners of a square, provide more than the absolute minimum number of points required for camera calibration.

where **A** is a $2n \times 11$ matrix of coefficients, which multiplies the G-matrix, now in the form:

$$\mathbf{g} = (G_{11}G_{12}G_{13}G_{14}G_{21}G_{22}G_{23}G_{24}G_{41}G_{42}G_{43})^{\mathrm{T}} \tag{19.25}$$

and $\boldsymbol{\xi}$ is a $2n$-element column vector of image coordinates. The pseudoinverse solution is:

$$\mathbf{g} = \mathbf{A}^{\dagger}\boldsymbol{\xi} \tag{19.26}$$

where

$$\mathbf{A}^{\dagger} = (\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathrm{T}} \tag{19.27}$$

The solution is more complex than might have been expected, as a normal matrix inverse is only defined, and can only be computed, for a square matrix. Note that solutions are only obtainable by this method if the matrix $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ is invertible. For further details of this method, see Golub and van Loan (1983).

19.4 INTRINSIC AND EXTRINSIC PARAMETERS

At this point, it is useful to look in more detail at the general transformation leading to camera calibration. When we are calibrating the camera, we are actually trying to bring the camera and world coordinate systems into coincidence. The first step is to move the origin of the world coordinates to the origin of the camera coordinate system. The second step is to rotate the world coordinate system until its axes are coincident with those of the camera coordinate system. The third step is to move the image plane laterally until there is complete agreement between the two coordinate systems: this step is required as it is not known initially which point in the world coordinate system corresponds to the principal point in the image. [The *principal point* is the image point lying on the principal axis of the camera: it is the point in the image which is closest to the center of projection. Correspondingly, the *principal axis* (or *optical axis*) of the camera is the line through the center of projection normal to the image plane.]

There is an important point to be borne in mind during this process. If the camera coordinates are given by **C**, then the translation **T** required in the first step will be—**C**. Similarly, the rotations that are required will be the inverses of those which correspond to the actual camera orientations. The reason for these reversals is that (for example) rotating an object (here the camera) forwards gives the same effect as rotating the axes backwards. Thus, all operations have to be carried out with the reverse arguments to those indicated above in Section 19.1. The complete transformation for camera calibration is hence:

$$\mathbf{G} = \mathbf{PLRT}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & t_1 \\ 0 & 1 & 0 & t_2 \\ 0 & 0 & 1 & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & 0 \\ R_{21} & R_{22} & R_{23} & 0 \\ R_{31} & R_{32} & R_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & T_1 \\ 0 & 1 & 0 & T_2 \\ 0 & 0 & 1 & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{19.28}$$

where matrix **P** takes account of the perspective transformation required to form the image. In fact, it is usual to group together the transformations **P** and **L** and call them internal camera transformations which include the *intrinsic camera parameters*, whereas **R** and **T** are taken together as external camera transformations corresponding to *extrinsic camera parameters*:

$$\mathbf{G} = \mathbf{G}_{\text{internal}}\mathbf{G}_{\text{external}} \tag{19.29}$$

where

$$\mathbf{G}_{\text{internal}} = \mathbf{PL} = \begin{bmatrix} 1 & 0 & 0 & t_1 \\ 0 & 1 & 0 & t_2 \\ 0 & 0 & 1 & t_3 \\ 0 & 0 & 1/f & t_3/f \end{bmatrix} \to \begin{bmatrix} 1 & 0 & t_1 \\ 0 & 1 & t_2 \\ 0 & 0 & 1/f \end{bmatrix} \tag{19.30}$$

$$\mathbf{G}_{\text{external}} = \mathbf{RT} = \begin{bmatrix} \mathbf{R_1} & \mathbf{R_1 \cdot T} \\ \mathbf{R_2} & \mathbf{R_2 \cdot T} \\ \mathbf{R_3} & \mathbf{R_3 \cdot T} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \tag{19.31}$$

In the matrix for $\mathbf{G}_{\text{internal}}$, we have assumed that the initial translation matrix $\mathbf{T}$ moves the camera's center of projection to the correct position, so that the value of t_3 can be made equal to zero: in that case, the effect of $\mathbf{L}$ will indeed be lateral as indicated above. At that point, we can express the (2-D) result in terms of a 3×3 homogeneous coordinate matrix. In the matrix for $\mathbf{G}_{\text{external}}$, we have expressed the result succinctly in terms of the rows $\mathbf{R}_1$, $\mathbf{R}_2$, $\mathbf{R}_3$ of $\mathbf{R}$, and have taken dot products with $\mathbf{T}$: the (3-D) result is a 4×4 homogenous coordinate matrix.

Although the above treatment gives a good indication of the underlying meaning of $\mathbf{G}$, it is not general because we have not so far included scaling and skew parameters in the internal matrix. In fact, the generalized form of $\mathbf{G}_{\text{internal}}$ is:

$$\mathbf{G}_{\text{internal}} = \begin{bmatrix} s_1 & b_1 & t_1 \\ b_2 & s_2 & t_2 \\ 0 & 0 & 1/f \end{bmatrix} \tag{19.32}$$

Potentially, $\mathbf{G}_{\text{internal}}$ should include the following:

1. A transform for correcting scaling errors.
2. A transform for correcting translation errors. (For this purpose, the origin of the image should be on the principal axis of the camera. Misalignment of the sensor may prevent this point from being at the center of the image.)
3. A transform for correcting sensor skewing errors (due to nonorthogonality of the sensor axes).
4. A transform for correcting sensor shearing errors (due to unequal scaling along the sensor axes).
5. A transform for correcting for unknown sensor orientation within the image plane.

Clearly, translation errors (Item 2) are corrected by adjusting t_1 and t_2. All the other adjustments are concerned with the values of the 2×2 submatrix:

$$\begin{bmatrix} s_1 & b_1 \\ b_2 & s_2 \end{bmatrix}$$

However, note that application of this matrix performs rotation within the image plane immediately after rotation has been performed in the world coordinates by $\mathbf{G}_{\text{external}}$, and it is virtually impossible to separate the two rotations. This explains why we now have a total of 6 external and 6 internal parameters totaling 12 rather than the expected 11 parameters (we return to the factor $1/f$ below). As a result, it is better to exclude Item 5 in the above list of internal transforms and to subsume it into the external parameters. (While doing so may not be ideal, there is no way of separating the two rotational components by purely optical means: only measurements on the internal dimensions of the camera system could determine the internal component, but separation is not likely to be a cogent or even meaningful matter. On the other hand, the internal component is likely to be stable, whereas the external component may be prone to variation if the camera is not mounted securely.) As the rotational component in $\mathbf{G}_{\text{internal}}$ has been excluded, b_1 and b_2 must now be equal, and the internal parameters will be: s_1, s_2, b, t_1, t_2. Note that the factor $1/f$ provides a scaling which cannot be separated from the other scaling factors during camera calibration, without specific (i.e., separate) measurement of f. Thus, we have a total of 6 parameters from $\mathbf{G}_{\text{external}}$ and 5 parameters from $\mathbf{G}_{\text{internal}}$: this totals 11 and equals the number cited in the previous section.

We next consider the special case where the sensor is known to be Euclidean to a high degree of accuracy. This will mean that $b = b_1 = b_2 = 0$, and $s_1 = s_2$, bringing the number of internal parameters down to three. In addition, if care has been taken over sensor alignment, and there are no other offsets to be allowed for, it may be known that $t_1 = t_2 = 0$. This will bring the total number of internal parameters down to just one, namely $s = s_1 = s_2$, or $s\,f$, if we take proper account of the focal length. In this case, there will be a total of seven calibration parameters for the whole camera system, and this may permit it to be set up unambiguously by viewing a known object having four clearly marked features instead of the six that would normally be required (see Section 19.3).

19.5 CORRECTING FOR RADIAL DISTORTIONS

Photographs generally appear so distortion free that there is a tendency to imagine that camera lenses are virtually perfect. However, it sometimes happens that a photograph will show odd curvatures of straight lines, particularly those appearing around the periphery of the picture. The results commonly take the form of "pincushion" or "barrel" distortion: these terms arise because pincushions have a tendency to be overextended at the corners, whereas barrels usually bulge in the middle. In images of paving stones or brick walls, the amount of distortion is usually not more than a few pixels in a total of the order of 512, i.e., typically less than 2%, and this explains why in the absence of particular straight line markers such distortions can be missed (Fig. 19.3). However, it is important both for recognition and for interimage matching purposes that any distortions should be eliminated. Indeed, image interpretation is nowadays targeted at, and frequently achieves, subpixel accuracy. In addition, disparities between stereo images are in the first order of small quantities, and single pixel errors would lead to significant errors in depth measurement. Hence, it is more the rule than the exception that 3-D image analysis will need to make corrections for barrel or pincushion distortion.

For reasons of symmetry, the distortions that arise in images tend to involve radial expansions or contractions relative to the optical axis—corresponding respectively to pincushion or barrel distortion. As with many types of error, series solutions can be useful. Thus, it is worthwhile to model the distortions as

$$\mathbf{r}' = \mathbf{r}f(r) = \mathbf{r}\left(a_0 + a_2 r^2 + a_4 r^4 + a_6 r^6 + \cdots\right) \tag{19.33}$$

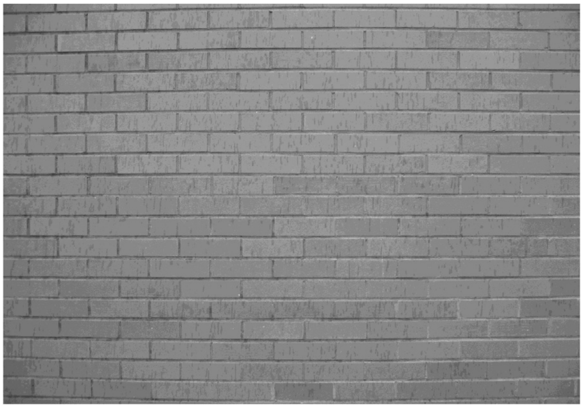

FIGURE 19.3

Photograph of a brick wall showing radial (barrel) distortion.

the odd orders in the brackets canceling to zero, again for reasons of symmetry. It is usual to set a_0 to unity, as this coefficient can be taken up by the scale parameters in the camera calibration matrix.

To fully define the effect, we write the x and y distortions as

$$x' - x_c = (x - x_c)\left(1 + a_2r^2 + a_4r^4 + a_6r^6 + \cdots\right) \tag{19.34}$$

$$y' - y_c = (y - y_c)\left(1 + a_2r^2 + a_4r^4 + a_6r^6 + \cdots\right) \tag{19.35}$$

Here, x and y are measured relative to the position of the optical axis of the lens (x_c, y_c), so $\mathbf{r} = (x{-}x_c, y{-}y_c)$, $\mathbf{r}' = (x'{-}x_c, y'{-}y_c)$.

As remarked above, the errors to be expected are in the range 2% or less. This means that it is normally sufficiently accurate to take just the first correction term in the expansion and disregard the rest. At the very least, this will introduce such a large improvement in the accuracy that it will be difficult to detect any discrepancies, especially if the image dimensions are 512 × 512 pixels or less. (This remark will not apply to many web cameras, which are sold at extremely low prices on the mainly amateur market. Although the camera chip and electronics are often very good value, the accompanying low-cost lens may well require extensive correction to ensure that distortion-free measurements are possible.) In addition, computation errors in matrix inversion and convergence of 3-D algorithms will add to the digitization errors, tending further to hide higher powers of radial distortion. Thus, in most cases, the latter can be modeled using a single parameter equation:

$$\mathbf{r}' = \mathbf{r}f(r) = \mathbf{r}\left(1 + a_2r^2\right) \tag{19.36}$$

Note that the above theory only models the distortion: clearly, it has to be corrected by the corresponding inverse transformation.

It is instructive to consider the apparent shape of a straight line which appears, for example, along the top of an image (Fig. 19.3). Take the image dimensions to range over $-x_1 \le x \le x_1$, $-y_1 \le y \le y_1$, and the optical axis of the camera to be at the center of the image. Then, the straight line will have the approximate equation:

$$y' = y_1\left[1 + a_2\left(x^2 + y_1^2\right)\right] = y_1 + a_2 y_1^3 + a_2 y_1 x^2 \tag{19.37}$$

which represents a parabola. The vertical error at the center of the parabola is $a_2 y_1^3$, and the additional vertical error at the ends is $a_2 y_1 x_1^2$. If the image is square ($x_1 = y_1$), these two errors are equal (the erroneous impression is given by the parabola shape that the error at $x = 0$ is zero).

Finally, note that digital scanners are very different from single lens cameras, in that their lenses travel along the object space during acquisition. Thus, longitudinal errors are unlikely to arise to anything like the same extent, though lateral errors could in principle be problematic.

19.6 MULTIPLE VIEW VISION

Over the 1990s, a considerable advance in 3-D vision was made by examining what could be learnt from uncalibrated cameras using multiple views. At first sight, considering the efforts made in earlier sections of this chapter to understand exactly how cameras should be calibrated, this may seem nonsensical. Nevertheless, there are considerable potential advantages in examining multiple views—not least, many thousands of videotapes are available from uncalibrated cameras, including those used for surveillance and those produced in the film industry. In such cases, as much must be made of the available material as possible, whether or not any regrets over “what might have been” are entertained. However, the need is deeper than this. Many situations exist in which the camera parameters might vary because of thermal variations, or because the zoom or focus setting has been adjusted: and it is impracticable to keep recalibrating a camera using accurately made test objects. Finally, if multiple (e.g., stereo) cameras are used, each will have to be calibrated separately, and the results compared to minimize the combined error: far better to examine the system as a whole and to calibrate it on the real scenes that are being viewed.

In fact, we have already met some aspects of these aspirations, in the form of invariants that are obtained in sequence by a single camera. For example, if a series of four collinear points are viewed and their cross ratio is checked, it will be found to be constant as the camera moves forward, changes orientation, or views the points increasingly obliquely—so long as they all remain within the field of view. For this purpose, all that is required to perform the recognition and maintain awareness of the object (the four points) is an uncalibrated but distortion-free camera. By distortion-free, we here mean not the ability to correct perspective distortion—which is, after all, the function of the cross ratio invariant—but the lack of radial distortion, or at least the capability in the following software for eliminating it (see Section 19.5).

To understand how image interpretation can be carried out more generally, using multiple views—whether from the same camera moved to a variety of places, or multiple cameras with overlapping views of the world—we shall need to go back to basics and start afresh with a more general attack on concepts such as binocular vision and epipolar constraints. In particular, two important matrices will be called into play—the "essential" matrix and the "fundamental" matrix. We start with the essential matrix and then generalize the idea to the fundamental matrix. But first, we need to look at the geometry of two cameras with general views of the world.

19.7 GENERALIZED EPIPOLAR GEOMETRY

In Section 16.3, we considered the stereo correspondence problem and had already simplified the task by choosing two cameras whose image planes were not only parallel but in the same plane. This made the geometry of depth perception especially simple, but suppressed possibilities allowed for in the human visual system (HVS), of having a nonzero vergence angle between the two images. Indeed, the HVS is special in adjusting vergence so that the current focus of attention in the field of view has almost zero disparity between the two images, and it seems likely that the HVS estimates depth not merely by measuring disparity but rather by measuring the vergence together with remanent small variations in disparity.

Here, we generalize the situation to cover the possibility of disparity coupled with substantial vergence. Fig. 19.4 shows the revised geometry. Note first that observation of a real point P in the scene leads to points P_1 and P_2 in the two images; that P_1 could correspond to any point on the epipolar line E_2 in Image 2; and similarly, that point P_2 could correspond to any point on the epipolar line E_1 in Image 1. Indeed, the so-called epipolar plane of P is the plane containing P and the projection points C_1 and C_2 of the two cameras: the epipolar lines (see Section 16.3) are thus the straight lines in which this plane cuts the two image planes. Furthermore, the line joining C_1 and C_2 cuts the image planes in the so-called epipoles e_1 and e_2: these can be regarded as the images of the alternate camera projection points. Note that all epipolar planes pass through points C_1, C_2 and e_1, e_2: this means that all epipolar lines in the two images pass through the respective epipoles. However, if the vergence angle were zero (as in Fig. 16.5), the epipoles would be at infinity in either direction, and all epipolar lines in either image would be parallel, and indeed parallel to the vector $\mathbf{C}$ from C_1 to C_2.

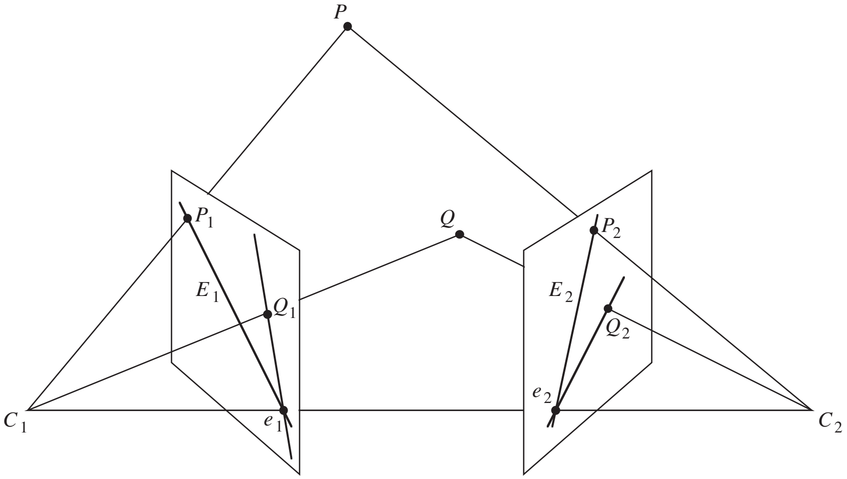

FIGURE 19.4

Generalized imaging of a scene from two viewpoints. In this case, there is substantial vergence. All epipolar lines in the left image pass through epipole e_1: of these, only E_1 is shown. Similar comments apply for the right image.

19.8 THE ESSENTIAL MATRIX

In this section, we start with the vectors $\mathbf{P}_1$, $\mathbf{P}_2$, from C_1, C_2 to P, and also the vector $\mathbf{C}$ from C_1 to C_2. Vector subtraction gives:

$$\mathbf{P}_2 = \mathbf{P}_1 - \mathbf{C} \tag{19.38}$$

We also know that $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{C}$ are coplanar, the condition of coplanarity being:

$$\mathbf{P}_2 \cdot \mathbf{C} \times \mathbf{P}_1 = 0 \tag{19.39}$$

(This can be thought of as bringing to zero the volume of the parallelepiped with sides $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{C}$.)

To progress, we need to relate the vectors $\mathbf{P}_1$ and $\mathbf{P}_2$ when these are expressed relative to their own frames of reference. If we take these vectors as having been defined in the C_1 frame of reference, we now reexpress $\mathbf{P}_2$ in its own (C_2) frame of reference, by applying a translation $\mathbf{C}$ and a rotation of coordinates expressed as the orthogonal matrix R. This leads to:

$$\mathbf{P}'_2 = R\mathbf{P}_2 = R(\mathbf{P}_1 - \mathbf{C}) \tag{19.40}$$

so that:

$$\mathbf{P}_2 = R^{-1}\mathbf{P}'_2 = R^{\mathrm{T}}\mathbf{P}'_2 \tag{19.41}$$

Substituting in the coplanarity condition gives:

$$\left(R^{\mathrm{T}}\mathbf{P}'_2\right)\cdot\mathbf{C}\times\mathbf{P}_1=0 \tag{19.42}$$

At this point, it is useful to replace the vector product notation by using a skew-symmetric matrix $C_\times$ to denote $\mathbf{C}\ \times$, where:

$$C_\times=\begin{bmatrix}0 & -C_z & C_y\\ C_z & 0 & -C_x\\ -C_y & C_x & 0\end{bmatrix} \tag{19.43}$$

At the same time, we observe the correct matrix formulation of all the vectors by transposing appropriately. We now find that:

$$\left(R^{\mathrm{T}}\mathbf{P}'_2\right)^{\mathrm{T}}C_\times\mathbf{P}_1=0 \tag{19.44}$$

$$\therefore\quad \mathbf{P}'^{\mathrm{T}}_2RC_\times\mathbf{P}_1=0 \tag{19.45}$$

Finally, we obtain the "essential matrix" formulation:

$$\mathbf{P}'^{\mathrm{T}}_2E\mathbf{P}_1=0 \tag{19.46}$$

where the essential matrix has been found to be:

$$E=RC_\times \tag{19.47}$$

Eq. (19.46) is actually the desired result: it expresses the relation between the observed positions of the same point in the two camera frames of reference. Furthermore, it immediately leads to formulae for the epipolar lines. To see this, first note that in the C_1 camera frame:

$$\mathbf{p}_1=\left(f_1/Z_1\right)\mathbf{P}_1 \tag{19.48}$$

whereas in the C_2 camera frame (and expressed in terms of that frame of reference):

$$\mathbf{p}'_2=\left(f_2/Z_2\right)\mathbf{P}'_2 \tag{19.49}$$

Eliminating $\mathbf{P}_1$ and $\mathbf{P}'_2$, and dropping the prime (as within the respective image planes the numbers 1 and 2 are sufficient to specify the coordinates unambiguously), we find:

$$\mathbf{p}_2^{\mathrm{T}}E\mathbf{p}_1=0 \tag{19.50}$$

as Z_1, Z_2 and f_1, f_2 can be canceled from this matrix equation.

Now note that writing $\mathbf{p}_2^{\mathrm{T}}E=\mathbf{l}_1^{\mathrm{T}}$ and $\mathbf{l}_2=E\mathbf{p}_1$ leads to the following relations:

$$\mathbf{p}_1^{\mathrm{T}}\mathbf{l}_1=0 \tag{19.51}$$

$$\mathbf{p}_2^{\mathrm{T}}\mathbf{l}_2=0 \tag{19.52}$$

This means that $\mathbf{l}_2 = E\mathbf{p}_1$ and $\mathbf{l}_1 = E^{\mathrm{T}}\mathbf{p}_2$ are the epipolar lines corresponding to $\mathbf{p}_1$ and $\mathbf{p}_2$, respectively. (To fully understand this, consider a line $\mathbf{l}$ and a point $\mathbf{p}$: $\mathbf{p}^{\mathrm{T}}\mathbf{l} = 0$ means that $\mathbf{p}$ lies on the line $\mathbf{l}$, or dually, $\mathbf{l}$ passes through the point $\mathbf{p}$.)

Finally, we can find the epipoles from the above formulation. In fact, the epipole lies on every epipolar line within the same image. Thus, $\mathbf{e}_2$ satisfies (can be substituted for $\mathbf{p}_2$ in) Eq. (19.52), and hence:

$$\begin{aligned} & \mathbf{e}_2^{\mathrm{T}}\mathbf{l}_2 = 0 \\ \therefore\quad & \mathbf{e}_2^{\mathrm{T}}E\mathbf{p}_1 = 0 \;\text{ for all } \mathbf{p}_1. \end{aligned}$$

This means that $\mathbf{e}_2^{\mathrm{T}}E = 0$, i.e., $E^{\mathrm{T}}\mathbf{e}_2 = 0$. Similarly, $E\mathbf{e}_1 = 0$.

19.9 THE FUNDAMENTAL MATRIX

Notice that in the last part of the essential matrix calculation, we implicitly assumed that the cameras are correctly calibrated. Specifically, $\mathbf{p}_1$ and $\mathbf{p}_2$ are corrected (calibrated) image coordinates. However, there is a need to work with uncalibrated images, using the raw pixel measurements —for all the reasons given in Section 19.6. (Note also that any radial distortions need to be eliminated, so as to idealize the camera, but not to calibrate it in the sense of Sections 19.3 and 19.4.) Applying the camera intrinsic matrices G_1, G_2 to the calibrated image coordinates (Section 19.4), we get the raw image coordinates:

$$\mathbf{q}_1 = G_1\mathbf{p}_1 \tag{19.53}$$

$$\mathbf{q}_2 = G_2\mathbf{p}_2 \tag{19.54}$$

In fact, we here need to go in the reverse direction, so we use the inverse equations:

$$\mathbf{p}_1 = G_1^{-1}\mathbf{q}_1 \tag{19.55}$$

$$\mathbf{p}_2 = G_2^{-1}\mathbf{q}_2 \tag{19.56}$$

Substituting for $\mathbf{p}_1$ and $\mathbf{p}_2$ in Eq. (19.50), we find the desired equation linking the raw pixel coordinates:

$$\mathbf{q}_2^{\mathrm{T}}\left(G_2{}^{-1}\right)^{\mathrm{T}}EG_1{}^{-1}\mathbf{q}_1 = 0 \tag{19.57}$$

which can be expressed as

$$\mathbf{q}_2^{\mathrm{T}}F\mathbf{q}_1 = 0 \tag{19.58}$$

where

$$F = \left(G_2{}^{-1}\right)^{\mathrm{T}}EG_1{}^{-1} \tag{19.59}$$

F is defined as the "fundamental matrix." Because it contains all the information that would be needed to calibrate the cameras, it contains more free parameters than the essential matrix. However, in other respects, the two matrices are intended to convey the same basic information, as is confirmed by the resemblance between the two formulations—Eqs. (19.46) and (19.58).

Finally, just as in the case of the essential matrix, the epipoles are given by $F\mathbf{f}_1 = 0$ and $F^{\mathrm{T}}\mathbf{f}_2 = 0$, though this time in raw image coordinates $\mathbf{f}_1$ and $\mathbf{f}_2$.

19.10 PROPERTIES OF THE ESSENTIAL AND FUNDAMENTAL MATRICES

Next, we consider the composition of the essential and fundamental matrices. In particular, note that $C_\times$ is a factor of E and also, indirectly, of F. In fact, they are homogeneous in $C_\times$, so the scale of $\mathbf{C}$ will make no difference to the two matrix formulations (Eqs. (19.46) and (19.58)), only the *direction* of $\mathbf{C}$ being important: indeed, the scales of both E and F are immaterial, and as a result, only the relative values of their coefficients are of importance. This means that there are at most only eight independent coefficients in E and F. In fact, in the case of F, there are only *seven*, as $C_\times$ is skew-symmetric, and this ensures that it has Rank 2 rather than Rank 3—a property that is passed on to F. The same argument applies for E, but the lower complexity of E (by virtue of its not containing the image calibration information) means that it has only *five* free parameters. In the latter case, it is easy to see what they are: they arise from the original three translation ($\mathbf{C}$) and three rotation ($\mathbf{R}$) parameters, but the scale parameter is excluded.

In this context, note that if $\mathbf{C}$ arises from a translation of a single camera, the same essential matrix will result whatever the scale of $\mathbf{C}$: only the direction of $\mathbf{C}$ actually matters, and the same epipolar lines will result from continued motion in the same direction. In fact, in this case, we can interpret the epipoles as foci of expansion or contraction. This underlines the power of this formulation: specifically, it treats motion and displacement a single entity.

Finally, we should try to understand why there are seven free parameters in the fundamental matrix. The solution is relatively simple. Each epipole requires two parameters to specify it. In addition, three parameters are needed to map any three epipolar lines from one image to the other. But why do just three epipolar lines have to be mapped? This is because the family of epipolar lines is a pencil whose orientations are related by cross ratios, so once three epipolar lines have been specified, the mapping of any other can be deduced. (Knowing the properties of the cross ratio, it is seen that fewer than three epipolar lines would be insufficient, and that more than three would yield no additional information.) This fact is sometimes stated in the following form: a homography (a projective transformation) between two 1-D projective spaces has three degrees of freedom.

19.11 ESTIMATING THE FUNDAMENTAL MATRIX

In the previous section, we showed that the fundamental matrix has seven free parameters. This means that it ought to be possible to estimate it by identifying the same seven features in the two images. However, using the minimum number of points in this way carries the health warning that they must be in general position: special configurations of points can lead to numerical instabilities in the computations, total failure to converge, or unnecessary ambiguities in the results. In general, coplanar points are to be avoided. In any case, although this is mathematically possible in principle, and a suitable nonlinear algorithm has been devised by Faugeras et al. (1992) to implement it, it has been shown that the computation can be numerically unstable. Essentially, noise acts as an additional variable boosting the effective number of degrees of freedom in the problem to eight. However, a linear algorithm called the *eight-point algorithm* has been devised to overcome the problem. Curiously, this algorithm had been proposed many years earlier by Longuet-Higgins (1981) to estimate the *essential* matrix, but it came into its own when Hartley (1995) showed how to control the errors by first normalizing the values. In addition, by using more than eight points, increased accuracy can be attained, but then a suitable algorithm must be found that can cope with the now overdetermined parameters. Principal component analysis can be used for this, an appropriate procedure being singular value decomposition.

Apart from noise, gross mismatches in forming trial point correspondences between images can be a source of practical problems. If so, the normal least squares types of solution can profitably be replaced by the least median of squares robust estimation method (Appendix A).

19.12 AN UPDATE ON THE EIGHT-POINT ALGORITHM

Section 19.11 outlined the value of the eight-point algorithm for estimating the fundamental matrix. Over a period of about 8 years (1995–2003), this essentially became the standard solution to the problem. However, a key contribution by Torr and Fitzgibbon (2003, 2004) has shown that the eight-point algorithm might after all not be the best possible method, as the solutions it obtains depend on the particular coordinate system used for the computation. This is because the normalization normally used, namely $\Sigma_i f_i^2 = 1$, is not invariant to shifts in the coordinate system. In fact, it is by no means obvious how to find an invariant normalization: note, for example, that the simple normalization $f_9 = 1$ suggested by Tsai and Huang (1984) leads to biassed solutions and excludes those with $f_9 = 0$. Nevertheless, Torr and Fitzgibbon's logical analysis of the situation, in which they were forced to disregard the affine transform case appropriate for weak perspective, led to the following normalization of F:

$$f_1^2 + f_2{}^2 + f_4{}^2 + f_5{}^2 = K \tag{19.60}$$

where K is a constant and:

$$F = \begin{bmatrix} f_1 & f_2 & f_3 \\ f_4 & f_5 & f_6 \\ f_7 & f_8 & f_9 \end{bmatrix} \tag{19.61}$$

Finally, to determine F, Eq. (19.60) can be applied as a Lagrangian multiplier constraint, and this leads to an eigenvector solution for F. Overall, the 8 × 8 eigenvalue problem solved by the eight-point algorithm is replaced by a 5 × 5 eigenvalue problem. Furthermore, this approach not only yields the required invariance properties, thus ensuring a more accurate solution, but also it gives a much faster computation that loses significantly fewer tracks in image sequence analysis.

19.13 IMAGE RECTIFICATION

In Section 19.7, we took some pains to generalize the epipolar approach and subsequently arrived at general solutions, corresponding to arbitrary overlapping views of scenes. However, there are distinct advantages in special views obtained from cameras with parallel axes—as in the case of Fig. 16.5 where the vergence is zero. Specifically, it is easier to find correspondences between scenes that are closely related in this way. Unfortunately, such well-prepared pairs of images are not in keeping with the aims promoted in Section 19.6, of insisting on closely aligned and calibrated cameras, and this certainly doesn't apply to frames taken by a single moving camera unless its motion is severely constrained by special means. In fact, the solution is straightforward: take images with uncalibrated cameras, estimate the fundamental matrix, and then apply suitable linear transformations to compute the images for any desired idealized camera positions. The latter technique is called image rectification and ensures for example, that the epipolar lines are all parallel to the baseline **C** between the centers of projection. This then results in correspondences being found by searching along points with the same ordinate in the alternate image: for a point with coordinates (x_1, y_1) in the first image, search for a matching point (x_2, y_1) in the second image.

When rectifying an image, it will in general be rotated in 3-D, and the obvious way of achieving this is to transfer each individual pixel to its new location in the rectified image. (Of course, it may also be translated and scaled, in which case the effect described here may be even more significant.). However, rotations are nonlinear processes and will in some cases have the effect of mapping several pixels into a single pixel; furthermore, a number of pixels may well not have intensity values assigned to them. While the first of these problems could be tackled by some sort of intensity averaging process, and the latter problem could be tackled by applying a median or other type of filter to the transformed image, such techniques are insufficiently thoroughgoing to provide accurate, reliable solutions. The *proper* way of overcoming these intrinsic difficulties is to backproject the pixel locations from the transformed image space to the source image, use interpolation to compute the ideal pixel intensities, and then transfer these intensities to the transformed image space.

Bilinear interpolation is used most often in the transformation process. This works by performing interpolation in the x-direction and then in the y-direction. Thus, if the location to be interpolated to is $(x + a, y + b)$ where x and y are integer pixel locations, and $0 \leq a, \ b \leq 1$, then the interpolated intensities in the x-direction are as follows:

$$I(x + a, y) = (1 - a)I(x, y) + aI(x + 1, y) \tag{19.62}$$

$$I(x + a,\ y + 1) = (1 - a)I(x,\ y + 1) + aI(x + 1,\ y + 1) \tag{19.63}$$

and the final result after interpolating in the y-direction is as follows:

$$\begin{aligned} I(x + a,\ y + b) = (1 - a)(1 - b)I(x,\ y) + a(1 - b)I(x + 1,\ y) \\ + (1 - a)bI(x, y + 1) + abI(x + 1,\ y + 1) \end{aligned} \tag{19.64}$$

The symmetry of the result shows that it makes no difference which axis is chosen for the first pair of interpolations, and this limits the arbitrariness of the method. Note that the method does not assume a locally planar intensity variation in 2-D: this is clear as the value of the $I(x + 1,\ y + 1)$ intensity is taken into account as well as the other three intensity values. Nevertheless, bilinear interpolation is not a totally ideal solution, as it takes no account of the sampling theorem, and for this reason, the bi-cubic interpolation method (which involves more computation) is sometimes used instead. In addition, all such methods introduce slight local blurring of the image as they involve averaging of local intensity values. Overall, transformation processes such as this are bound to result in slight degradation of the image data.

19.14 3-D RECONSTRUCTION

In Section 19.10, the fact that F is determined only up to an unknown scale factor (or equivalently that the actual scales of its coefficients as obtained are arbitrary) was strongly emphasized. This reflected the deliberate avoidance of camera calibration in this work. In practice, this means that if the results of computations of F are to be related back to the real world, the scaling factor must be reinstated. In principle, this can be achieved by viewing a single yardstick: it is unnecessary to view an object such as a Rubik cube, as knowledge of F carries with it a lot of information on relative dimensions in the real world. This factor is important when reconstructing a real scene with a real depth map.

There are a number of methods for image reconstruction, of which perhaps the most obvious is triangulation. This starts by taking two camera positions containing normalized images and projecting rays for a given point P back into the real world until they meet. In fact, attempting to do this meets with an immediate problem: the inaccuracies in the available parameters, coupled with the pixellation of the images, ensure that in most cases rays will not actually meet, as they are skew lines. The best that can be done with skew lines is to determine the position of closest approach. Once this has been found, the bisector of the line of closest approach (which is perpendicular to each of the rays) is, in *this* model, the most accurate estimate of the position of P in space.

Unfortunately, the above model is not guaranteed to give the most accurate prediction of the position of P. This is because perspective projection is a highly nonlinear process: in particular, slight misjudgment of the orientation of the point from either of the images can cause a substantial depth error, coupled with a significant lateral error: so much is indeed obvious from Fig. 19.5. This being so, it has to be asked where the error might still be linear, so that, at that position at least, error calculation can be based on Gaussian distributions. (Here, we ignore the possibility of gross errors arising from mismatches between images, which is

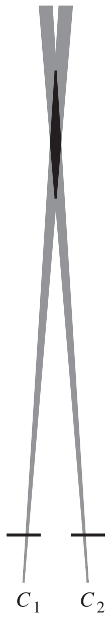

FIGURE 19.5

Error in locating a feature in space using binocular imaging. The dark-shaded regions represent the regions of space that could arise for small errors in the image planes. The crossover region, shaded black, confirms that longitudinal errors will be much larger than lateral errors. A full analysis would involve applying Gaussian or other error functions (see text).

the subject of further discussion in Section 19.11 and elsewhere.) In fact, the errors can be taken to be approximately Gaussian in the images themselves. This means that the point in space that has to be chosen as representing the most accurate interpretation of the data is that which results in the minimum error (in a least mean square sense) when reprojected onto the image planes. Typically, the error obtained using this approach is a factor of two smaller than that for the triangulation method described above (Hartley and Zisserman, 2000).

Finally, it is useful to mention a further type of error that can arise with two cameras. This applies when they both view an object with a smoothly varying boundary. For example, if both cameras are viewing the right hand edge of a vase of circular cross section, each will see a different point on the boundary and a discrepancy will arise in the estimated boundary position (Fig. 19.6). It is left as an

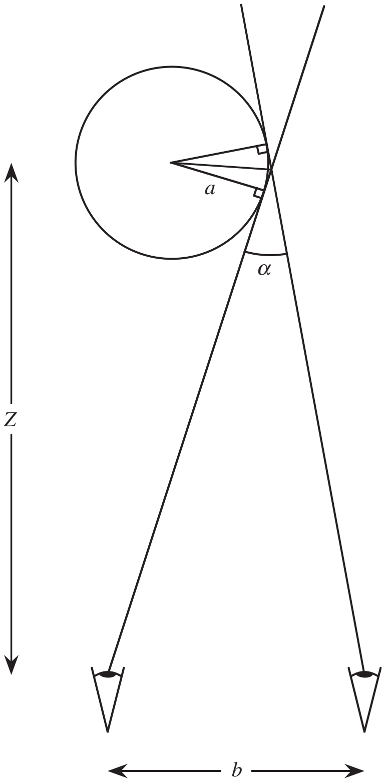

FIGURE 19.6

Lateral estimation error arising with a smoothly varying boundary. The error arises in estimating the boundary position when information from two views is fused in the standard way. *a* is the radius of a vase being observed, α is the disparity in direction of its right hand boundary, *Z* is its depth in the scene, and *b* is the stereo baseline.

exercise (Section 19.17) to determine the exact magnitude of such errors. In fact, the error is proportional both to a, where a is the local radius of curvature of the observed boundary, and to Z^{-2}, where Z is the depth in the scene. This means that the error (and the percentage error) tends to zero at large distances, and also that the error falls properly to zero for sharp corners.

19.15 CONCLUDING REMARKS

This chapter has discussed the transformations required for camera calibration and has outlined how calibration can be achieved. The camera parameters have been classified as "internal" and "external," thereby simplifying the conceptual problem and throwing light on the origins of errors in the system. It has been shown that a minimum of six points is required to perform calibration in the general case where 11 transformation parameters are involved; however, the number of points required might be reduced somewhat in special cases, e.g., where the sensor is known to be Euclidean. Nevertheless, it is normally more important to increase the number of points used for calibration than to attempt to reduce it, as substantial gains in accuracy can be obtained via the resulting averaging process.

In an apparent break with the previous work, Section 19.5 introduced multiple view vision. This important topic was seen to rest on generalized epipolar geometry and led to the essential and fundamental matrix formulations, which relate the observed positions of any point in two camera frames of reference. The importance of the eight-point algorithm for estimating either of these matrices—and particularly the fundamental matrix, which is relevant when the cameras are uncalibrated—was stressed. In addition, the need for accuracy in estimating the fundamental matrix is still a research issue.

The obvious way of tackling vision problems is to set up a camera and calibrate it, and only then to use it in anger. This chapter has shown how, to a large extent, calibration can be avoided or carried out adaptively "on the fly"—by performing multiple view vision and analyzing the various key matrices that arise from the generalized epipolar problems.

19.16 BIBLIOGRAPHICAL AND HISTORICAL NOTES

One of the first to use the various transformations described in this chapter was Roberts (1965). Important early references for camera calibration are the *Manual of Photogrammetry* (Slama, 1980), Tsai and Huang (1984), and Tsai (1986). Tsai's paper is especially useful in that he provides an extended, highly effective treatment which copes with nonlinear lens distortions. More recent papers on this topic include Haralick (1989), Crowley et al. (1993), Cumani and Guiducci

(1995), and Robert (1996): see also Zhang (1995). Note that parametrized plane curves can be used instead of points for the purpose of camera calibration (Haralick and Chu, 1984).

Clearly, camera calibration is an old topic that is revisited every time 3-D vision has to be used for measurement, and otherwise when rigorous analysis of 3-D scenes is called for. The calibration scenario started to undergo a metamorphosis in the early 1990s, when it was realized that much could be learnt without overt calibration, but rather by *comparing* images taken from moving sequences or from multiple views (Faugeras, 1992; Faugeras et al., 1992; Hartley, 1992; Maybank and Faugeras, 1992). In fact, although it was appreciated that much could be learnt without overt calibration, it was not at that stage known how much *might* be learnt, and there ensued a rapid sequence of developments as the frontiers were progressively pushed back (e.g., Hartley, 1995; Hartley, 1997; Luong and Faugeras, 1997). By the late 1990s, the fast evolution phase was over, and definitive, albeit quite complex, texts appeared covering these developments (Hartley and Zisserman, 2000; Faugeras and Luong, 2001; Gruen and Huang, 2001). Nevertheless, many refinements of the standard methods were still emerging (Faugeras et al., 2000; Heikkilä, 2000; Sturm, 2000; Roth and Whitehead, 2002). It is in this light that the innovative insights of Torr and Fitzgibbon (2003, 2004) and Chojnacki et al. (2003) expressing similar but not identical sentiments relating to the eight-point algorithm should be considered.

In retrospect, it is amusing that the early, incisive paper by Longuet-Higgins (1981) presaged many of these developments: although his eight-point algorithm applied specifically to the essential matrix, it was only very much later (Faugeras, 1992; Hartley, 1992) that it was applied to the fundamental matrix, and even later, in a crucial step, that its accuracy was greatly improved by prenormalizing the image data (Hartley, 1997). As already noted, the eight-point algorithm continued to be a focus for new research.

19.16.1 MORE RECENT DEVELOPMENTS

Most recently, Gallo et al. (2011) have studied how planes may be fitted to surfaces that are obtained from range data (i.e., sets of data points whose real-world (X, Y, Z) coordinates are approximately known). Although RANSAC should provide useful solutions, it sometimes fails when finding pairs of planar patches, and a single plane is fitted to both, with the result that it contains more inliers than the correct models. To cope with this, they devised an alternate form of random sample consensus (RANSAC), connected components-RANSAC (CC-RANSAC), which only considers the largest connected components of inliers for a given plane hypothesis. The method requires an inlier threshold to be set, and this has to be adjusted for the particular application in question. One relevant application is automatic car parking where a single level near a curb has to be identified.

Although the 8-point algorithm has become standard for solving the fundamental matrix, the latter only contains seven independent parameters so only identifying the same seven features in two images should be enough to solve it. Bartoli and Sturm (2004) have found that this is realizable if nonlinear estimation is used. The method converges faster than other approaches, though it is somewhat more likely to fall into local minima than methods based on redundant parameters. Fathy et al. (2011) study error criteria for fundamental matrix estimation. They show that the symmetric epipolar distance criterion is biased and find that of a number of available criteria, the recently developed Kanatani distance criterion (Kanatani et al., 2008) appears to be the most accurate. Ansar and Daniilidis (2003) have devised a novel set of algorithms for linear pose estimation from n points or n lines. The methods will find solutions for cases of $n \geq 4$, for points in general position. Although two similar existing noniterative methods exist in the case of estimation from n points (to which the new method is shown to be superior), there is no directly competing case for n lines.

19.17 PROBLEMS

1. For a 2-camera stereo system, obtain a formula for the depth error that arises for a given error in disparity. Hence, show that the percentage error in depth is numerically equal to the percentage error in disparity. What does this result mean in practical terms? How does the pixellation of the image affect the result?
2. A cylindrical vase with a circular cross section of local radius a is viewed by two cameras (Fig. 19.6). Obtain a formula giving the error δ in the estimated position of the boundary of the vase. Simplify the calculation by assuming that the boundary is on the perpendicular bisector of the line joining the centers of projection of the two cameras, and hence find α (Fig. 19.6) in terms of b and Z. Determine δ in terms of α and then substitute for α from the previous formula. Hence, justify the statements made at the end of Section 19.14.
3. Discuss the potential advantages of trinocular vision in the light of the theory of Section 19.8. What would be the best placement for a third camera? Where should the third camera *not* be placed? Would any gain be achieved by incorporating even more views of a scene?