for the purpose of reconstructing surfaces. They need textured images to work well. However, due to foreshortening effects and change in illumination direction, they are inadequate for matching image pairs taken from very different viewpoints. Also, the interpolation necessary to refine correspondences from pixel to subpixel precision can make correlation-based matching quite expensive.[7]

Feature-based methods are suitable when *a priori* information is available about the scene, so that optimal features can be used. A typical example is the case of indoor scenes, which usually contain many straight lines but rather untextured surfaces. Feature-based algorithms can also prove faster than correlation-based ones, but any comparison of specific algorithms must take into account the cost of producing the feature descriptors. The sparse disparity maps generated by these methods may look inferior to the dense maps of correlation-based matching, but in some applications (e.g., visual navigation) they may well be all you need in order to perform the required tasks successfully. Another advantage of feature-based techniques is that they are relatively insensitive to illumination changes and highlights.

The performance of any correspondence methods is jeopardised by *occlusions* (points with no counterpart in the other image) and *spurious matches* (false corresponding pairs created by noise). Appropriate constraints reduce the effects of both phenomena: two important ones are the *left-right consistency constraint* (only corresponding pairs found matching left-to-right *and* right-to-left are accepted), and the *epipolar constraint*, explained in the next section.

7.3 Epipolar Geometry

We now move on to study the geometry of stereo in its full generality. This will enable us to clarify what information is needed in order to perform the search for corresponding elements only along image lines. First of all, we need to establish some basic notations.

7.3.1 Notation

The geometry of stereo, known as *epipolar geometry*, is shown in Figure 7.6. The figure shows two pinhole cameras, their projection centers, O_l and O_r, and image planes, π_l and π_r. The focal lengths are denoted by f_l and f_r. As usual, each camera identifies a 3-D reference frame, the origin of which coincides with the projection center, and the Z-axis with the optical axis. The vectors $\mathbf{P}_l = [X_l, Y_l, Z_l]^\top$ and $\mathbf{P}_r = [X_r, Y_r, Z_r]^\top$ refer to the same 3-D point, P, thought of as a vector in the left and right camera reference frames respectively (Figure 7.6). The vectors $\mathbf{p}_l = [x_l, y_l, z_l]^\top$ and $\mathbf{p}_r = [x_r, y_r, z_r]^\top$ refer to the projections of P onto the left and right image plane respectively, and are expressed in the corresponding reference frame (Figure 7.6). Clearly, for all the image points we have $z_l = f_l$ or $z_r = f_r$, according to the image. Since each image plane can be thought of as a subset of the projective space P^2, image points can be equivalently thought of as points of the projective space P^2 (see Appendix, section A.4).

[7] One of the projects suggested at the end of this chapter deals with a parsimonious implementation of correlation-based matching.

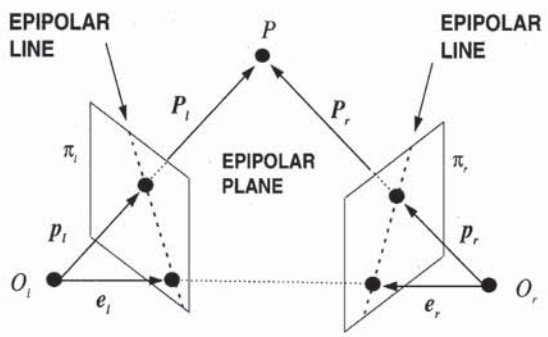

Figure 7.6 The epipolar geometry.

☞ Note that point vectors denoted by the *same* bold capital letter but by *different* subscripts, like $\mathbf{P}_l$ and $\mathbf{P}_r$ identify the *same point* in space. The subscript l or r tells you *the reference frame in which the vectors are expressed* (left or right). Instead, point vectors denoted by the *same* bold small letter but by a *different* subscript, like $\mathbf{p}_l$ and $\mathbf{p}_r$, identify *different points* in space (i.e., belonging to different image planes). In this case, the subscript tells you *also* the image plane on which the vectors lie. This is a slightly unfair but very effective abuse of notation.

7.3.2 Basics

The reference frames of the left and right cameras are related via the extrinsic parameters. These define a rigid transformation in 3-D space, defined by a translation vector, $\mathbf{T} = (O_r - O_l)$, and a rotation matrix, R. Given a point P in space, the relation between $\mathbf{P}_l$ and $\mathbf{P}_r$ is therefore

$$\mathbf{P}_r = R(\mathbf{P}_l - \mathbf{T}). \tag{7.7}$$

The name *epipolar geometry* is used because the points at which the line through the centers of projection intersects the image planes (Figure 7.6) are called *epipoles*. We denote the left and right epipole by $\mathbf{e}_l$ and $\mathbf{e}_r$ respectively. By construction, *the left epipole is the image of the projection center of the right camera and vice versa.*

☞ Notice that, if the line through the centers of projection is parallel to one of the image planes, the corresponding epipole is the point at infinity of that line.

The relation between a point in 3-D space and its projections is described by the usual equations of perspective projection, in vector form:

$$\mathbf{p}_l = \frac{f_l}{Z_l}\mathbf{P}_l \tag{7.8}$$

and

$$\mathbf{p}_r = \frac{f_r}{Z_r}\mathbf{P}_r. \tag{7.9}$$

The practical importance of epipolar geometry stems from the fact that the plane identified by P, O_l, and O_r, called *epipolar plane*, intersects each image in a line, called *epipolar line* (see Figure 7.6). Consider the triplet P, $\mathbf{p}_l$, and $\mathbf{p}_r$. Given $\mathbf{p}_l$, P can lie anywhere on the ray from O_l through $\mathbf{p}_l$. But, since the image of this ray in the right image is the epipolar line through the corresponding point, $\mathbf{p}_r$, *the correct match must lie on the epipolar line*. This important fact is known as the *epipolar constraint*. It establishes a mapping between points in the left image and lines in the right image and *vice versa*.

☞ Incidentally, since all rays include the projection center by construction, this also proves that all the epipolar lines go through the epipole.

So, if we determine the mapping between points on, say, the left image and corresponding epipolar lines on the right image, we can restrict the search for the match of $\mathbf{p}_l$ along the corresponding epipolar line. *The search for correspondences is thus reduced to a 1-D problem.* Alternatively, the same knowledge can be used to verify whether or not a candidate match lies on the corresponding epipolar line. This is usually a most effective procedure to *reject false matches* due to occlusions. Let us now summarize the main ideas encountered in this section:

Definition: Epipolar Geometry

Given a stereo pair of cameras, any point in 3-D space, P, defines a plane, π_P, going through P and the centers of projection of the two cameras. The plane π_P is called *epipolar plane*, and the lines where π_P intersects the image planes *conjugated epipolar lines*. The image in one camera of the projection center of the other is called *epipole*.

Properties of the Epipoles

With the exception of the epipole, only one epipolar line goes through any image point.
All the epipolar lines of one camera go through the camera's epipole.

Definition: Epipolar Constraint

Corresponding points must lie on conjugated epipolar lines.

The obvious question at this point is, can we estimate the epipolar geometry? Or equivalently, how do we determine the mapping between points in one image and epipolar lines in the other? This is the next problem we consider. Its solution also makes clear the relevance of epipolar geometry for reconstruction.

7.3.3 The Essential Matrix, E

The equation of the epipolar plane through P can be written as the coplanarity condition of the vectors $\mathbf{P}_l$, $\mathbf{T}$, and $\mathbf{P}_l - \mathbf{T}$ (Figure 7.6), or

$$(\mathbf{P}_l - \mathbf{T})^\top \mathbf{T} \times \mathbf{P}_l = 0.$$

Using (7.7), we obtain

$$(R^\top \mathbf{P}_r)^\top \mathbf{T} \times \mathbf{P}_l = 0. \tag{7.10}$$

Recalling that a vector product can be written as a multiplication by a rank-deficient matrix, we can write

$$\mathbf{T} \times \mathbf{P}_l = S\mathbf{P}_l$$

where

$$S = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix}. \tag{7.11}$$

Using this fact, (7.10) becomes

$$\mathbf{P}_r^\top E \mathbf{P}_l = 0, \tag{7.12}$$

with

$$E = RS. \tag{7.13}$$

Note that, by construction, S has always rank 2. The matrix E is called the *essential matrix* and *establishes a natural link between the epipolar constraint and the extrinsic parameters of the stereo system*. You will learn how to recover the extrinsic parameters from the essential matrix in the next section. In the meantime, observe that, using (7.8) and (7.9), and dividing by $Z_r Z_l$, (7.12) can be rewritten as

$$\mathbf{p}_r^\top E \mathbf{p}_l = 0. \tag{7.14}$$

As already mentioned, the image points $\mathbf{p}_l$ and $\mathbf{p}_r$, which lie on the left and right image planes respectively, can be regarded as points in the projective plane P^2 defined by the left and right image planes respectively (Appendix, section A.4). Consequently, you are entitled to think of $E\mathbf{p}_l$ in (7.14) as the projective line in the right plane, $\mathbf{u}_r$, that goes through $\mathbf{p}_r$ and the epipole $\mathbf{e}_r$:

$$\mathbf{u}_r = E\mathbf{p}_l. \tag{7.15}$$

As shown by (7.14) and (7.15), *the essential matrix is the mapping between points and epipolar lines we were looking for*.

☞ Notice that the whole discussion used coordinates in the *camera* reference frame, but what we actually measure from images are pixel coordinates. Therefore, in order to be able to make profitable use of the essential matrix, we need to know the transformation from *camera coordinates* to *pixel coordinates*, that is, the intrinsic parameters. This limitation is removed in the next section, but at a price.

7.3.4 The Fundamental Matrix, F

We now show that the mapping between points and epipolar lines can be obtained from corresponding points only, *with no prior information on the stereo system*.

Let M_l and M_r be the matrices of the intrinsic parameters (Chapter 2) of the left and right camera respectively. If $\tilde{\mathbf{p}}_l$ and $\tilde{\mathbf{p}}_r$ are the points in *pixel* coordinates corresponding to $\mathbf{p}_l$ and $\mathbf{p}_r$ in camera coordinates, we have

$$\mathbf{p}_l = M_l^{-1}\tilde{\mathbf{p}}_l \tag{7.16}$$

and

$$\mathbf{p}_r = M_r^{-1}\tilde{\mathbf{p}}_r. \tag{7.17}$$

By substituting (7.16) and (7.17) into (7.14), we have

$$\tilde{\mathbf{p}}_r^\top F \tilde{\mathbf{p}}_l = 0, \tag{7.18}$$

where

$$F = M_r^{-\top} E M_l^{-1}. \tag{7.19}$$

F is named *fundamental matrix*. The essential and fundamental matrix, as well as (7.14) and (7.18), are formally very similar. As with $E\mathbf{p}_l$ in (7.14), $F\tilde{\mathbf{p}}_l$ in (7.18) can be thought of as the equation of the projective epipolar line, $\tilde{\mathbf{u}}_r$, that correspond to the point $\tilde{\mathbf{p}}_l$, or

$$\tilde{\mathbf{u}}_r = F\tilde{\mathbf{p}}_l. \tag{7.20}$$

The most important difference between (7.15) and (7.20), and between the essential and fundamental matrices, is that *the fundamental matrix is defined in terms of pixel coordinates, the essential matrix in terms of camera coordinates.* Consequently, if you can estimate the fundamental matrix from a number of point matches in pixel coordinates, *you can reconstruct the epipolar geometry with no information at all on the intrinsic or extrinsic parameters*.

☞ This indicates that the epipolar constraint, as the mapping between points and corresponding epipolar lines, can be established with *no* prior knowledge of the stereo parameters.

The definitions and basic mathematical properties of these two important matrices are worth a summary.

Definition: Essential and Fundamental Matrices

For each pair of corresponding points $\mathbf{p}_l$ and $\mathbf{p}_r$ in *camera* coordinates, the *essential matrix* satisfies the equation

$$\mathbf{p}_r^\top E \mathbf{p}_l = 0.$$

For each pair of corresponding points $\tilde{\mathbf{p}}_l$ and $\tilde{\mathbf{p}}_r$ in *pixel* coordinates, the *fundamental matrix* satisfies the equation

$$\tilde{\mathbf{p}}_r^\top F \tilde{\mathbf{p}}_l = 0.$$

Properties

Both matrices enable full reconstruction of the epipolar geometry.

If M_l and M_r are the matrices of the intrinsic parameters, the relation between the essential and fundamental matrices is given by

$$F = M_r^{-T} E M_l^{-1}.$$

The essential matrix:

1. encodes information on the extrinsic parameters only (see (7.13))
2. has rank 2, since S in (7.11) has rank 2 and R full rank
3. its two nonzero singular values are equal

The fundamental matrix:

1. encodes information on both the intrinsic and extrinsic parameters
2. has rank 2, since T_l and T_r have full rank and E has rank 2

7.3.5 Computing E and F: The Eight-point Algorithm

How do we compute the essential and fundamental matrices? Of the various methods possible, the eight-point algorithm is by far the simplest and definitely the one you cannot ignore (if you are curious about other techniques, look into the Further Readings). We consider here the fundamental matrix only, and leave it to you to work out the straightforward modification needed to recover the essential matrix.

The idea behind the eight-point algorithm is very simple. Assume that you have been able to establish n point correspondences between the images. Each correspondence gives you a homogeneous linear equation like (7.18) for the nine entries of F; these equations form a homogeneous linear system. If you have at least eight correspondences (i.e., $n \geq 8$) and the n points do not form degenerate configurations,[8] the nine entries of F can be determined as the nontrivial solution of the system. Since the system is homogeneous, the solution is unique up to a signed scaling factor. If one uses more than eight points, so that the system is overdetermined, the solution can once again be obtained by means of SVD related techniques. If A is the system's matrix and $A = UDV^\top$, the solution is the column of V corresponding to the only null singular value of A (see Appendix, section A.6),

☞ Because of noise, numerical errors and inaccurate correspondences, A is more likely to be full rank, and the solution is the column of V associated with the *least* singular value of A.

☞ The estimated fundamental matrix is almost certainly nonsingular. We can enforce the singularity constraint by adjusting the entries of the estimated matrix F as done in Chapter 6

[8] For a thorough discussion of the degenerate configurations of eight or more points, as well as of the instabilities in the estimation of the essential and fundamental matrices, see the Further Readings.

for rotation matrices: we compute the singular value decomposition of the estimated matrix, $\hat{F} = UDV^\top$, and set the smallest singular value on the diagonal of the matrix D equal to 0. If D' is the corrected D matrix, the corrected estimate, F' is given by $F' = UD'V^\top$ (see Appendix, section A.6).

The following is the basic structure of the eight-point algorithm:

Algorithm EIGHT_POINT

The input is formed by n point correspondences, with $n \geq 8$.

1. Construct system (7.18) from n correspondences. Let A be the $n \times 9$ matrix of the coefficients of the system and $A = UDV^\top$ the SVD of A.
2. The entries of F (up to an unknown, signed scale factor) are the components of the column of V corresponding to the least singular value of A.
3. To enforce the singularity constraint, compute the singular value decomposition of F:
$$F = UDV^\top.$$
4. Set the smallest singular value in the diagonal of D equal to 0; let D' be the corrected matrix.
5. The corrected estimate of F, F', is finally given by
$$F' = UD'V^\top.$$

The output is the estimate of the fundamental matrix, F'.

☞ In order to avoid numerical instabilities, the eight-point algorithm should be implemented with care. The most important action to take is *to normalize the coordinates of the corresponding points so that the entries of A are of comparable size*. Typically, the first two coordinates (in pixels) of an image point are referred to the top left corner of the image, and can vary between a few pixels to a few hundreds; the differences can make A seriously ill-conditioned (Appendix, section A.6). To make things worse, the third (homogeneous) coordinate of image points is usually set to one. A simple procedure to avoid numerical instability is to translate the first two coordinates of each point to the centroid of each data set, and scale the norm of each point so that the average norm over the data set is 1. This can be accomplished by multiplying each left (right) point by two suitable 3×3 matrices, H_l and H_r (see Exercise 7.6 for details on how to compute both H_l and H_r). The algorithm EIGHT_POINT is then used to estimate the matrix $\tilde{F} = H_r F H_l$, and F obtained as $H_r^{-1} \tilde{F} H_l^{-1}$.

7.3.6 Locating the Epipoles from E and F

We can now establish the relation between the epipoles and the two matrices E *and* F. Consider for example the fundamental matrix, F. Since $\tilde{\mathbf{e}}_l$ lies on all the epipolar lines of the left image, we can rewrite (7.18) as

$$\tilde{\mathbf{p}}_r^\top F \tilde{\mathbf{e}}_l = 0$$

for every $\tilde{\mathbf{p}}_r$. But since F is not identically zero, this is possible if and only if

$$F\tilde{\mathbf{e}}_l = 0. \tag{7.21}$$

From (7.21) and the fact that F has rank 2, it follows that *the epipole, $\tilde{\mathbf{e}}_l$, is the null space of F*; similarly, $\tilde{\mathbf{e}}_r$ is the null space of $F^\top$.

We are now in a position to present an algorithm for finding the epipoles. Accurate epipole localization is helpful for refining the location of corresponding epipolar lines, checking the geometric consistency of the entire construction, simplifying the stereo geometry, and recovering 3-D structure in the case of uncalibrated stereo.

Again we present the algorithm in the case of the fundamental matrix. The adaptation to the case of the essential matrix is even simpler than before. The algorithm follows easily from (7.21): To determine the location of the epipoles, it is sufficient to find the null spaces of F and $F^\top$.

☞ These can be determined, for instance, from the singular value decomposition $F = UDV^\top$ and $F^\top = VDU^\top$ as column of V and U respectively corresponding to the null singular value in the diagonal matrix D.

Algorithm EPIPOLES_LOCATION

The input is the fundamental matrix F.

1. Find the SVD of F, that is, $F = UDV^\top$.
2. The epipole $\mathbf{e}_l$ is the column of V corresponding to the null singular value.
3. The epipole $\mathbf{e}_r$ is the column of U corresponding to the null singular value.

The output are the epipoles, $\mathbf{e}_l$ and $\mathbf{e}_r$.

☞ Notice that we can safely assume that there is exactly one singular value equal to 0 because algorithm EIGHT_POINT enforces the singularity constraint explicitly.

It has to be noticed that there are alternative methods to locate the epipoles, not based on the fundamental matrix and requiring as few as 6 point correspondences. More about them in the Further Readings.

7.3.7 Rectification

Before moving on to the problem of 3-D reconstruction, we want to address the issue of *rectification*. Given a pair of stereo images, rectification determines a transformation (or *warping*) of each image such that *pairs of conjugate epipolar lines become collinear and parallel to one of the image axes*, usually the horizontal one. Figure 7.7 shows an example. The importance of rectification is that the correspondence problem, which involves 2-D search in general, *is reduced to a 1-D search on a scanline identified trivially*. In other words, to find the point corresponding to (i_l, j_l) of the left image, we just look along the scanline $j = j_l$ in the right image.

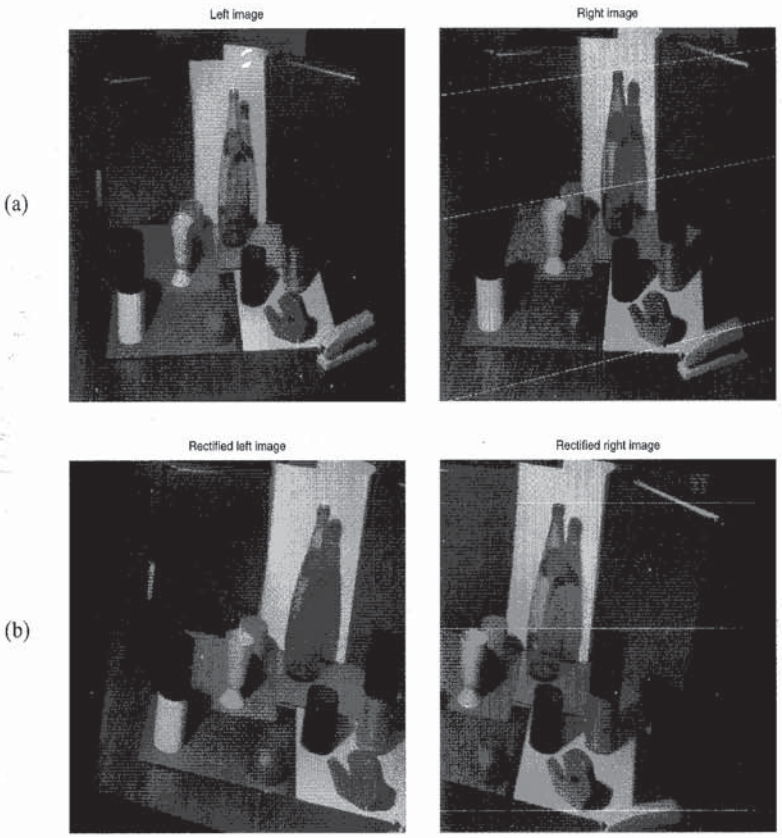

Figure 7.7 (a) A stereo pair. (b) The pair rectified. The left images plot the epipolar lines corresponding to the points marked in the right pictures. Stereo pair courtesy of INRIA (France).

Let us begin by stating the problem and our assumptions.

Assumptions and Problem Statement

Given a stereo pair of images, the intrinsic parameters of each camera, and the extrinsic parameters of the system, R and $\mathbf{T}$, compute the image transformation that makes conjugated epipolar lines collinear and parallel to the horizontal image axis.

The assumption of knowing the intrinsic and extrinsic parameters is not strictly necessary (see Further Readings) but leads to a very simple technique. How do we

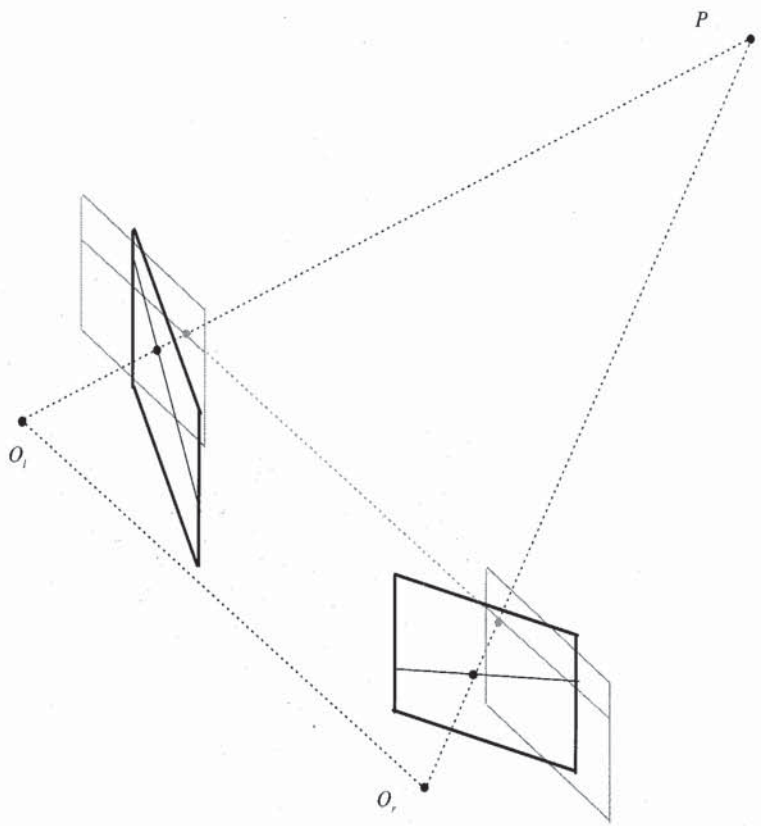

Figure 7.8 Rectification of a stereo pair. The epipolar lines associated to a 3-D point P in the original cameras (black lines) become collinear in the rectified cameras (light grey). Notice that the original cameras can be in any position, and the optical axes may not intersect.

go about computing the rectifying image transformation? The rectified images can be thought of as acquired by a new stereo rig, obtained by rotating the original cameras around their optical centers. This is illustrated in Figure 7.8, which shows also how the points of the rectified images are determined from the points of the original images and their corresponding projection rays.

We proceed to describe a rectification algorithm assuming, without losing generality, that in both cameras

1. the origin of the image reference frame is the principal point;
2. the focal length is equal to f.

The algorithm consists of four steps:

- Rotate the left camera so that the epipole goes to infinity along the horizontal axis.
- Apply the same rotation to the right camera to recover the original geometry.
- Rotate the right camera by R.
- Adjust the scale in both camera reference frames.

To carry out this method, we construct a triple of mutually orthogonal unit vectors $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$. Since the problem is underconstrained, we are going to make an arbitrary choice. The first vector, $\mathbf{e}_1$, is given by the epipole; since the image center is in the origin, $\mathbf{e}_1$ coincides with the direction of translation, or

$$\mathbf{e}_1 = \frac{\mathbf{T}}{\|\mathbf{T}\|}.$$

The only constraint we have on the second vector, $\mathbf{e}_2$, is that it must be orthogonal to $\mathbf{e}_1$. To this purpose, we compute and normalize the cross product of $\mathbf{e}_1$ with the direction vector of the optical axis, to obtain

$$\mathbf{e}_2 = \frac{1}{\sqrt{T_x^2 + T_y^2}} [-T_y, T_x, 0]^\top.$$

The third unit vector is unambiguously determined as

$$\mathbf{e}_3 = \mathbf{e}_1 \times \mathbf{e}_2.$$

It is easy to check that the orthogonal matrix defined as

$$R_{rect} = \begin{pmatrix} \mathbf{e}_1^\top \\ \mathbf{e}_2^\top \\ \mathbf{e}_3^\top \end{pmatrix} \tag{7.22}$$

rotates the left camera about the projection center in such a way that the epipolar lines become parallel to the horizontal axis. This implements the first step of the algorithm. Since the remaining steps are straightforward, we proceed to give the customary algorithm:

Algorithm RECTIFICATION

The input is formed by the intrinsic and extrinsic parameters of a stereo system and a set of points in each camera to be rectified (which could be the whole images). In addition, Assumptions 1 and 2 above hold.

1. Build the matrix R_{rect} as in (7.22);
2. Set $R_l = R_{rect}$ and $R_r = R R_{rect}$;

3. For each left-camera point, $\mathbf{p}_l = [x, y, f]^\top$ compute

$$R_l \mathbf{p}_l = [x', y', z']$$

and the coordinates of the corresponding rectified point, $\mathbf{p}'_l$, as

$$\mathbf{p}'_l = \frac{f}{z'}[x', y', z'].$$

4. Repeat the previous step for the right camera using R_r and $\mathbf{p}_r$.

The output is the pair of transformations to be applied to the two cameras in order to rectify the two input point sets, as well as the rectified sets of points.

Notice that the rectified coordinates are in general not integer. Therefore, if you want to obtain integer coordinates (for instance if you are rectifying the whole images), you should implement RECTIFICATION backwards, that is, starting from the *new* image plane and applying the *inverse* transformations, so that the pixel values in the *new* image plane can be computed as a bilinear interpolation of the pixel values in the *old* image plane.

☞ A rectified image is not in general contained in the same region of the image plane as the original image. You may have to alter the focal lengths of the rectified cameras to keep all the points within images of the same size as the original.

We are now fully equipped to deal with the reconstruction problem of stereo.

7.4 3-D Reconstruction

We have learned methods for solving the correspondence problem and determining the epipolar geometry from at least eight point correspondences. At this point, the 3-D reconstruction that can be obtained depends on the amount of *a priori* knowledge available on the parameters of the stereo system; we can identify three cases.[9] First, if both intrinsic and extrinsic parameters are known, you can solve the reconstruction problem unambiguously by triangulation, as detailed in section 7.1. Second, if only the intrinsic parameters are known, you can still solve the problem and, at the same time, estimate the extrinsic parameters of the system, but only *up to an unknown scaling factor*. Third, if the pixel correspondences are the only information available, and neither the intrinsic nor the extrinsic parameters are known, you can still obtain a reconstruction of the environment, but only *up to an unknown, global projective transformation*. Here is a visual summary.

[9] In reality there are several intermediate cases, but we concentrate on these three for simplicity.

A Priori Knowledge	3-D Reconstruction from Two Views
Intrinsic and extrinsic parameters	Unambiguous (absolute coordinates)
Intrinsic parameters only	Up to an unknown scaling factor
No information on parameters	Up to an unknown projective transformation of the environment

We now consider these three cases in turn.

7.4.1 Reconstruction by Triangulation

This is the simplest case. If you know both the intrinsic and the extrinsic parameters of your stereo system, reconstruction is straightforward.

Assumptions and Problem Statement

Under the assumption that the intrinsic and extrinsic parameters are known, compute the 3-D location of the points from their projections, $\mathbf{p}_l$ and $\mathbf{p}_r$.

As shown in Figure 7.6, the point P, projected into the pair of corresponding points $\mathbf{p}_l$ and $\mathbf{p}_r$, lies at the intersection of the two rays from O_l through $\mathbf{p}_l$ and from O_r through $\mathbf{p}_r$ respectively. In our assumptions, the rays are known and the intersection can be computed. The problem is, since parameters and image locations are known only approximately, *the two rays will not actually intersect in space*; their intersection can only be estimated as the point of minimum distance from both rays. This is what we set off to do.

Let $a\mathbf{p}_l$ ($a \in \mathbb{R}$) be the ray, l, through O_l and $\mathbf{p}_l$. Let $\mathbf{T} + bR^\top\mathbf{p}_r$ ($b \in \mathbb{R}$) be the ray, r, through O_r and $\mathbf{p}_r$ expressed in the left reference frame. Let $\mathbf{w}$ be a vector orthogonal to both l and r. Our problem reduces to determining the midpoint, P', of the segment parallel to $\mathbf{w}$ that joins l and r (Figure 7.9).

This is very simple because the endpoints of the segment, say $a_0\mathbf{p}_l$ and $\mathbf{T} + b_0R^\top\mathbf{p}_r$, can be computed solving the linear system of equations

$$a\mathbf{p}_l - bR^\top\mathbf{p}_r + c(\mathbf{p}_l \times R^\top\mathbf{p}_r) = \mathbf{T} \tag{7.23}$$

for a_0, b_0, and c_0. We summarize this simple method below:

Algorithm TRIANG

All vectors and coordinates are referred to the left camera reference frame. The input is formed by a set of corresponding points; let $\mathbf{p}_l$ and $\mathbf{p}_r$ be a generic pair.

Let $a\mathbf{p}_l$, $a \in \mathbb{R}$, be the ray l through O_l ($a = 0$) and $\mathbf{p}_l$ ($a = 1$). Let $\mathbf{T} + bR^\top\mathbf{p}_r$, $b \in \mathbb{R}$, the ray r through O_r ($b = 0$) and $\mathbf{p}_r$ ($b = 1$). Let $\mathbf{w} = \mathbf{p}_l \times R^\top\mathbf{p}_r$ the vector orthogonal to both l and r, and $a\mathbf{p}_l + c\mathbf{w}$, $c \in \mathbb{R}$, the line w through $a\mathbf{p}_l$ (for some fixed a) and parallel to $\mathbf{w}$.

1. Determine the endpoints of the segment, s, belonging to the line parallel to $\mathbf{w}$ that joins l and r, $a_0\mathbf{p}_l$ and $\mathbf{T} + b_0R^\top\mathbf{p}_r$, by solving (7.23).
2. The triangulated point, P', is the midpoint of the segment s.

The output is the set of reconstructed 3-D points.

☞ The determinant of the coefficients of system (7.23) is the triple product of $\mathbf{p}_l$, $R^\top\mathbf{p}_r$, and $\mathbf{p}_l \times R^\top\mathbf{p}_r$. Therefore, as expected from geometric considerations, the system has a unique solution if and only if the two rays l and r are not parallel.

☞ Reconstruction can be performed from rectified images directly; that is, without going back to the coordinate frames of the original pair (Exercise 7.7).

How often can we assume to know the intrinsic and extrinsic parameters of a stereo system? If the geometry of the system does not change with time, the intrinsic and extrinsic parameters of each camera can be estimated through the procedures of Chapter 6. If $\mathbf{T}_l$, R_l, and $\mathbf{T}_r$, R_r are the extrinsic parameters of the two cameras in the world reference frame, it is not difficult to show that the extrinsic parameters of the stereo system, $\mathbf{T}$ and R, are

$$\begin{aligned} R &= R_r R_l^\top \\ \mathbf{T} &= \mathbf{T}_l - R^\top\mathbf{T}_r. \end{aligned} \tag{7.24}$$

Try to derive (7.24) yourself. If you need help, see Exercise 7.10.

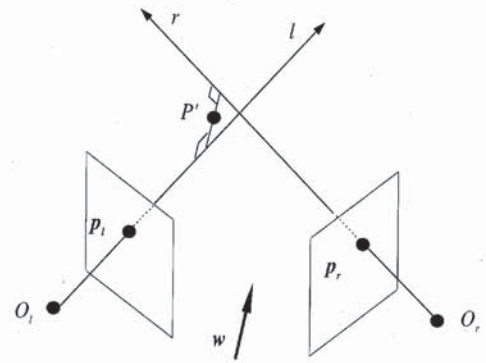

Figure 7.9 Triangulation with nonintersecting rays.

7.4.2 Reconstruction up to a Scale Factor

We now consider the case in which *only the intrinsic parameters of both the cameras are known* and derive a method to estimate the extrinsic parameters of the stereo system as well as the 3-D structure of the scene. Since the method makes use of the essential matrix, we must assume that at least eight point correspondences have been established.

Assumptions and Problem Statement

Assuming that only the intrinsic parameters and n point correspondences are given, with $n \geq 8$, compute the location of the 3-D points from their projections, $\mathbf{p}_l$ and $\mathbf{p}_r$.

Unlike triangulation, in which the geometry of the stereo system was fully known, the solution cannot rely on sufficient information to locate the 3-D points unambiguously. Intuitively, *since we do not know the baseline of the system, we cannot recover the true scale of the viewed scene*. Consequently, the reconstruction is unique only up to an unknown scaling factor. This factor can be determined if we know the distance between two points in the observed scene.

The origin of this ambiguity is quite clear in the method that we now present. The first step requires estimation of the essential matrix, E, which can only be known up to an arbitrary scale factor; therefore, we look for a convenient normalization of E. From the definition of the essential matrix, (7.13), we have

$$E^\top E = S^\top R^\top R S = S^\top S,$$

or

$$E^\top E = \begin{bmatrix} T_y^2 + T_z^2 & -T_x T_y & -T_x T_z \\ -T_y T_x & T_z^2 + T_x^2 & -T_y T_z \\ -T_z T_x & -T_z T_y & T_x^2 + T_y^2 \end{bmatrix}. \tag{7.25}$$

From (7.25) we have that the trace of $EE^\top$ is

$$Tr(E^\top E) = 2\|\mathbf{T}\|^2,$$

so that dividing the entries of the essential matrix by

$$N = \sqrt{Tr(E^\top E)/2}$$

is equivalent to normalizing the length of the translation vector to unit.

☞ Notice that, by effect of this normalization, the difference between the true essential matrix and the one estimated through the eight-point algorithm is, at most, a global sign change.

Using this normalization, (7.25) can be rewritten as

$$\hat{E}^\top \hat{E} = \begin{bmatrix} 1 - \hat{T}_x^2 & -\hat{T}_x \hat{T}_y & -\hat{T}_x \hat{T}_z \\ -\hat{T}_y \hat{T}_x & 1 - \hat{T}_y^2 & -\hat{T}_y \hat{T}_z \\ -\hat{T}_z \hat{T}_x & -\hat{T}_z \hat{T}_y & 1 - \hat{T}_z^2 \end{bmatrix}, \tag{7.26}$$

where $\hat{E}$ is the normalized essential matrix and $\hat{\mathbf{T}} = \mathbf{T}/\|\mathbf{T}\|$ the normalized translation vector. Recovering the components of $\hat{\mathbf{T}}$ from any row or column of the matrix $\hat{E}^\top \hat{E}$ is now a simple matter. However, since each entry of the matrix $\hat{E}^\top \hat{E}$ in (7.26) is quadratic in the components of $\hat{\mathbf{T}}$, the estimated components might differ from the true components by a global sign change. Let us assume, for the time being, that $\hat{\mathbf{T}}$ has been recovered with the proper global sign; then the rotation matrix can be obtained by simple algebraic computations. We define

$$\mathbf{w}_i = \hat{\mathbf{E}}_i \times \hat{\mathbf{T}}, \tag{7.27}$$

with $i = 1, 2, 3$ and $\hat{\mathbf{E}}_i$ the three rows of the normalized essential matrix $\hat{E}$, thought of as 3-D vectors. If $\mathbf{R}_i$ are the rows of the rotation matrix R, again thought of as 3-D vectors, easy but rather lengthy algebraic calculations yield

$$\mathbf{R}_i = \mathbf{w}_i + \mathbf{w}_j \times \mathbf{w}_k \tag{7.28}$$

with the triplet (i, j, k) spanning all cyclic permutations of $(1, 2, 3)$.

In summary, given an estimated, normalized essential matrix, we end up with four different estimates for the pair $(\hat{\mathbf{T}}, R)$. These four estimates are generated by the twofold ambiguity in the sign of $\hat{E}$ and $\hat{\mathbf{T}}$. The 3-D reconstruction of the viewed points resolves the ambiguity and finds the only correct estimate. For each of the four pairs $(\hat{\mathbf{T}}, R)$, we compute the third component of each point in the left camera reference frame. From (7.7) and (7.9), and since $Z_r = \mathbf{R}_3^\top (\mathbf{P}_l - \hat{\mathbf{T}})$, we obtain

$$\mathbf{p}_r = \frac{f_r R(\mathbf{P}_l - \hat{\mathbf{T}})}{\mathbf{R}_3^\top (\mathbf{P}_l - \hat{\mathbf{T}})}.$$

Thus, for the first component of $\mathbf{p}_r$ we have

$$x_r = \frac{f_r \mathbf{R}_1^\top (\mathbf{P}_l - \hat{\mathbf{T}})}{\mathbf{R}_3^\top (\mathbf{P}_l - \hat{\mathbf{T}})}. \tag{7.29}$$

Finally, plugging (7.8) into (7.29) with $\mathbf{T} = \hat{\mathbf{T}}$, and solving for Z_l,

$$Z_l = f_l \frac{(f_r \mathbf{R}_1 - x_r \mathbf{R}_3)^\top \hat{\mathbf{T}}}{(f_r \mathbf{R}_1 - x_r \mathbf{R}_3)^\top \mathbf{p}_l}. \tag{7.30}$$

We can recover the other coordinates of $\mathbf{P}_l$ from (7.8), and the coordinates of $\mathbf{P}_r$ from the relation

$$\mathbf{P}_r = R(\mathbf{P}_l - \hat{\mathbf{T}}). \tag{7.31}$$

It turns out that only one of the four estimates of $(\hat{\mathbf{T}}, R)$ yields geometrically consistent (i.e., positive) Z_l and Z_r coordinates for *all* the points. The actions to take in order to determine the correct solution are detailed in the box below, which summarizes the entire algorithm.

Algorithm EUCLID_REC

The input is formed by a set of corresponding image points in camera coordinates, with $\mathbf{p}_l$ and $\mathbf{p}_r$ a generic pair, and an estimate of the normalized essential matrix, $\hat{E}$.

1. Recover $\hat{\mathbf{T}}$ from (7.26).
2. Construct the vectors $\mathbf{w}$ from (7.27), and compute the rows of the matrix R through (7.28).
3. Reconstruct the Z_l and Z_r coordinates of each point using (7.30), (7.8) and (7.31).
4. If the signs of Z_l and Z_r of the reconstructed points are
   (a) both negative for some point, change the sign of $\hat{\mathbf{T}}$ and go to step 3;
   (b) one negative, one positive for some point, change the sign of each entry of $\hat{E}$ and go to step 2;
   (c) both positive for all points, exit.

The output is the set of reconstructed 3-D points (up to a scale factor).

☞ When implementing EUCLID_REC, make sure that the algorithm does not go through more than 4 iterations of steps 2-4 (since there are only 4 possible combinations for the unknown signs of $\hat{\mathbf{T}}$ and $\hat{E}$). Keep in mind that, in the case of very small displacements, the errors in the disparity estimates may be sufficient to make the 3-D reconstruction inconsistent; when this happens, the algorithm keeps going through steps 2-4.

7.4.3 Reconstruction up to a Projective Transformation

The aim of this section is to show that you can compute a 3-D reconstruction even in the absence of *any* information on the intrinsic and extrinsic parameters. The price to pay is that *the reconstruction is unique only up to an unknown projective transformation of the world*. The Further Readings point you to methods for determining this transformation.

Assumptions and Problem Statement

Assuming that only n point correspondences are given, with $n \geq 8$ (and therefore the location of the epipoles, $\mathbf{e}$ and $\mathbf{e}'$), compute the location of the 3-D points from their projections, $\mathbf{p}_l$ and $\mathbf{p}_r$.

☞ It is worth noticing that, if no estimates of the intrinsic and extrinsic parameters are available and nonlinear deformations can be neglected, the accuracy of the reconstruction is only affected by that of the algorithms computing the disparities, not by calibration.

The plan for this section is as follows. We show that, mapping five arbitrary scene points into the standard projective basis of P^3, and using the epipoles, the projection matrix of each camera can be explicitly recovered up to an unknown projective transformation (the one associating the standard basis to the five points selected, which is unknown as we do not know the location of the five 3-D points in camera coordinates).[10] Once the projection matrices are determined, the 3-D location of an arbitrary point in space is obtained by triangulation in projective space. You can find the essential notions of projective geometry needed to cope with all this in the Appendix, section A.4.

Determining the Projection Matrices. In order to carry out our plan, we introduce a slight change of notation. In what follows, we drop the l and r subscripts and adopt the unprimed and primed letters to indicate points in the left and right images respectively. In addition, capital letters now denote points in the projective space P^3 (four coordinates), while small letters points in P^2 (three coordinates). The 3-D space is regarded as a subset of P^3, and each image plane as a subset of P^2. This means that we regard the 3-D point $[X, Y, Z]^\top$ of $\mathbb{R}^3$ as the point $[X, Y, Z, 1]^\top$ of P^3, and a point $[x, y]^\top$ of $\mathbb{R}^2$ as the point $[x, y, 1]^\top$ of P^2. Let $\mathbf{O}$ and $\mathbf{O}'$ denote the projection centers.

We let $\mathbf{P}_1, \ldots, \mathbf{P}_n$ be the points in P^3 to be recovered from their left and right images, $\mathbf{p}_1, \ldots, \mathbf{p}_n$ and $\mathbf{p}'_1, \ldots, \mathbf{p}'_n$, and assume that, of the first five $\mathbf{P}_i$ $(\mathbf{P}_1, \mathbf{P}_2, \ldots \mathbf{P}_5)$, no three are collinear and no four are coplanar.

We first show that, if we choose $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_5$ as the standard projective basis of P^3 (see Appendix, section A.4), *each projection matrix can be determined up to a projective factor that depends on the location of the epipoles*. Since a *spatial projective transformation* is fixed if the destiny of five points is known, we can, without losing generality, set up a projective transformation that sends $\mathbf{P}_1, \mathbf{P}_2, \ldots \mathbf{P}_5$ into the standard projective basis of P^3, $\mathbf{P}_1 = [1, 0, 0, 0]^\top$, $\mathbf{P}_2 = [0, 1, 0, 0]^\top$, $\mathbf{P}_3 = [0, 0, 1, 0]^\top$, $\mathbf{P}_4 = [0, 0, 0, 1]^\top$, and $\mathbf{P}_5 = [1, 1, 1, 1]^\top$.

For the corresponding image points $\mathbf{p}_i$ in the left camera, we can write

$$M\mathbf{P}_i = \rho_i \mathbf{p}_i, \tag{7.32}$$

where M is the projection matrix and $\rho_i \neq 0$. Similarly, since a *planar projective transformation* is fixed if the destiny of four points is known, we can also set up a projective transformation that sends the first four $\mathbf{p}_i$ into the standard projective basis of P^2, that is, $\mathbf{p}_1 = (1, 0, 0)^\top$, $\mathbf{p}_2 = (0, 1, 0)^\top$, $\mathbf{p}_3 = (0, 0, 1)^\top$, and $\mathbf{p}_4 = (1, 1, 1)^\top$. In what follows, it is assumed that the coordinates of the fifth point, $\mathbf{p}_5$, of the epipole $\mathbf{e}$, and of any other image point, $\mathbf{p}_i$, are obtained applying this transformation to their *old* coordinates.

The purpose of all this is to simplify the expression of the projection matrix: substituting $\mathbf{P}_1, \ldots, \mathbf{P}_4$ and $\mathbf{p}_1, \ldots, \mathbf{p}_4$ into (7.32), we see that the matrix M can be rewritten as

$$M = \begin{bmatrix} \rho_1 & 0 & 0 & \rho_4 \\ 0 & \rho_2 & 0 & \rho_4 \\ 0 & 0 & \rho_3 & \rho_4 \end{bmatrix}. \tag{7.33}$$

[10] You should convince yourself that knowing the locations of the five points *in the camera reference frame* amounts to camera calibration, which rather defeats the point of uncalibrated stereo.

Let $[\alpha, \beta, \gamma]^\top$ be the coordinates of $\mathbf{p}_5$ in the standard basis; (7.32) with $i = 5$ makes it possible to eliminate ρ_1, ρ_2 and ρ_3 from (7.33), obtaining

$$M = \begin{bmatrix} \alpha\rho_5 - \rho_4 & 0 & 0 & \rho_4 \\ 0 & \beta\rho_5 - \rho_4 & 0 & \rho_4 \\ 0 & 0 & \gamma\rho_5 - \rho_4 & \rho_4 \end{bmatrix}. \tag{7.34}$$

Finally, since a projection matrix is defined only up to a scale factor, we can divide each entry of matrix (7.34) by ρ_4, obtaining

$$M = \begin{bmatrix} \alpha x - 1 & 0 & 0 & 1 \\ 0 & \beta x - 1 & 0 & 1 \\ 0 & 0 & \gamma x - 1 & 1 \end{bmatrix}. \tag{7.35}$$

where $x = \rho_5/\rho_4$. *The projection matrix of the left camera has been determined up to the unknown projective parameter x.*

In order to determine x, it is useful to relate the entries of M to the coordinates of the projection center, $\mathbf{O}$. This can be done by observing that M models a perspective projection with $\mathbf{O}$ as projection center. Therefore, M *projects* every point of P^3, with the exception of $\mathbf{O}$, into a point of P^2. Since M has rank 3, the null space of M is nontrivial and consists necessarily of $\mathbf{O}$:

$$M\mathbf{O} = 0. \tag{7.36}$$

Equation (7.36) can be solved for O_x, O_y and O_z:

$$\mathbf{O} = \left[\frac{1}{1-\alpha x}, \frac{1}{1-\beta x}, \frac{1}{1-\gamma x}, 1\right]^\top. \tag{7.37}$$

Corresponding relations and results can be obtained for the right camera (in the primed reference frame). In particular, we can write

$$M' = \begin{bmatrix} \alpha' x' - 1 & 0 & 0 & 1 \\ 0 & \beta' x' - 1 & 0 & 1 \\ 0 & 0 & \gamma' x' - 1 & 1 \end{bmatrix}.$$

and

$$\mathbf{O}' = \left[\frac{1}{1-\alpha' x'}, \frac{1}{1-\beta' x'}, \frac{1}{1-\gamma' x'}, 1\right]^\top. \tag{7.38}$$

Since the location of the epipoles is known, x and x' (and hence the full projection matrices and the centers of projection) can be determined from

$$M\mathbf{O}' = \sigma\mathbf{e} \tag{7.39}$$

and

$$M'\mathbf{O} = \sigma'\mathbf{e}' \tag{7.40}$$

with $\sigma \neq 0$ and $\sigma' \neq 0$.[11]

Let's see first what we can recover from (7.39). Substituting (7.35) and (7.38) into (7.39), we obtain the following system of equations

$$\begin{bmatrix} \alpha & -\alpha' & \alpha' e_x \\ \beta & -\beta' & \beta' e_y \\ \gamma & -\gamma' & \gamma' e_z \end{bmatrix} \begin{pmatrix} x \\ x' \\ \sigma x' \end{pmatrix} = \begin{pmatrix} \sigma e_x \\ \sigma e_y \\ \sigma e_z \end{pmatrix} \tag{7.41}$$

Since σ is unknown, system (7.41) is homogeneous and nonlinear in the three unknown x, x', and σ. However, we can regard it as a linear system in the unknown x, x' and $\sigma x'$, so that solving for $\sigma x'$ we have

$$\sigma x' = \sigma \frac{\mathbf{e}^\top(\mathbf{p}_5 \times \mathbf{p}'_5)}{\mathbf{v}^\top(\mathbf{p}_5 \times \mathbf{p}'_5)} \tag{7.42}$$

with $\mathbf{v} = (\alpha' e_x, \beta' e_y, \gamma' e_z)$. Since $\mathbf{e}$, $\mathbf{p}_5$, $\mathbf{p}'_5$, and $\mathbf{v}$ are known and the unknown factor σ cancels out, (7.42) actually determines x'.

A similar derivation applied to (7.40) yields

$$x = \frac{\mathbf{e}'^\top(\mathbf{p}_5 \times \mathbf{p}'_5)}{\mathbf{v}'^\top(\mathbf{p}_5 \times \mathbf{p}'_5)} \tag{7.43}$$

with $\mathbf{v}' = (\alpha e'_x, \beta e'_y, \gamma e'_z)$. Having determined both x and x' we can regard both the projection matrices and the centers of projections as completely determined.

Computing the Projective Reconstruction. We are now in a position to reconstruct *any* point in P^3 given its corresponding image points, $\mathbf{p} = [p_x, p_y, p_z]^\top$ and $\mathbf{p}' = [p'_x, p'_y, p'_z]^\top$. The reconstruction is unique up to the unknown projective transformation fixed by the choice of $\mathbf{P}_1, \ldots, \mathbf{P}_5$ as the standard basis for P^3. Observe that the projective line l defined by

$$\lambda\mathbf{O} + \mu\,[O_x p_x, O_y p_y, O_z p_z, 0]^\top, \tag{7.44}$$

with $\lambda, \mu \in \mathbb{R}$ and not both 0, goes through $\mathbf{O}$ (for $\lambda = 1$ and $\mu = 0$) and also through $\mathbf{p}$, since

$$M \begin{pmatrix} O_x p_x \\ O_y p_y \\ O_z p_z \\ 0 \end{pmatrix} = \mathbf{p}.$$

[11] Since the epipoles and the centers of projection lie on a straight line, (7.39) and (7.40) are not independent. For the purpose of this brief introduction, however, this can be safely ignored.

Similarly, the projective line l'

$$\lambda' \mathbf{O}' + \mu' \left[O'_x p'_x, O'_y p'_y, O'_z p'_z, 0 \right]^\top ,$$

with $\lambda', \mu' \in \mathbb{R}$ and not both 0, goes through $\mathbf{O}'$ and $\mathbf{p}'$. *The projective point* $\mathbf{P}$ *can thus be obtained by intersecting the two projective lines* l *and* l'. This amounts to looking for the non-trivial solution of the homogeneous system of linear equations

$$\begin{bmatrix} O_x & O_x p_x & -O'_x & -O'_x p'_x \\ O_y & O_y p_y & -O'_y & -O'_y p'_y \\ O_z & O_z p_z & -O'_z & -O'_z p'_z \\ 1 & 0 & -1 & 0 \end{bmatrix} \begin{pmatrix} \lambda \\ \mu \\ \lambda' \\ \mu' \end{pmatrix} = 0. \tag{7.45}$$

☞ Once again, singular value decomposition $UDV^\top$ of the system matrix of (7.45) provides a numerically stable procedure for solving this linear system: The solution is given by the column of V associated with the smallest singular value along the diagonal of D.

Algorithm UNCAL_STEREO

The input is formed by n pairs of corresponding points, $\mathbf{p}_i$ and $\mathbf{p}'_i$, with $i =, \ldots, n$ and $n \geq 5$, images of n points, $\mathbf{P}_1, \ldots, \mathbf{P}_n$. We assume that, of the first five $\mathbf{P}_i$ ($\mathbf{P}_1, \mathbf{P}_2, \ldots \mathbf{P}_5$), no three are collinear and no four are coplanar.

We assume to have estimated the location of the epipoles, $\mathbf{e}$ and $\mathbf{e}'$, using EPIPOLES_LOCATION. Let $\mathbf{P}_1, \ldots, \mathbf{P}_5$ be the standard projective basis of P^3. We assume the same notation used throughout the section.

1. Determine the planar projective transformations T and T' that map the $\mathbf{p}_i$ and $\mathbf{p}'_i$ ($i = 1, \ldots, 4$) into the standard projective basis of P^2 on each image plane. Apply T to the $\mathbf{p}_i$ and the epipole $\mathbf{e}$, and T' to the $\mathbf{p}'_i$ and the epipole $\mathbf{e}'$. Let (α, β, γ) and $(\alpha', \beta', \gamma')$ be the new coordinates of $\mathbf{p}_5$ and $\mathbf{p}'_5$.
2. Determine x and x' from (7.42) and (7.43).
3. Determine $\mathbf{O}$ and $\mathbf{O}'$ from (7.37) and (7.38).
4. Given a pair of corresponding points $\mathbf{p}$ and $\mathbf{p}'$, reconstruct the location of the point $\mathbf{P}$ in the standard projective basis of P^3 using (7.44) with λ and μ nontrivial solution of (7.45).

The output is formed by the coordinates of $\mathbf{P}_1, \ldots, \mathbf{P}_n$ in the standard projective basis.

Having found a projective reconstruction of our points, *how do we go back to Euclidean coordinates*? If we know the location of $\mathbf{P}_1, \ldots, \mathbf{P}_5$ in the world frame, we can determine the projective transformation introduced at the beginning of this section that mapped these five points, thought of as points of P^3, into the standard projective basis (see the Appendix, section A.4 for details on how to do it). The Further Readings point to (nontrivial) algorithms for Euclidean reconstruction which relax this assumption, but need more than two images.

7.5 Summary

After working through this chapter you should be able to:

- ❑ explain the fundamental concepts and problems of stereo
- ❑ solve the correspondence problem by means of correlation-based and feature-based techniques
- ❑ estimate the fundamental (and, if possible, the essential) matrix from point correspondences
- ❑ determine the epipolar geometry and rectifying a stereo pair
- ❑ recover 3-D structure from image correspondences when *(a)* both intrinsic and extrinsic parameters are known, *(b)* only the intrinsic parameters are known, and *(c)* both intrinsic and extrinsic parameters are unknown

7.6 Further Readings

The literature on stereo is immense. You may wish to start with the two classic correspondence algorithms by Marr and Poggio [14, 15]. Among the multitude of correspondence algorithms proposed in the last two decades, we suggest the methods proposed in [2, 9, 12, 17]. A way to adapt the shape and size of SSD windows to different image parts is described in Kanade and Okutomi [10].

Rectification is discussed by Ayache [1] and Faugeras [5]. A MATLAB implementation of a rectification algorithm based on projection matrices can be downloaded from `ftp://taras.dimi.uniud.it/pub/sources/rectif_m.tar.gz`. The algorithm and the implementation are due to Andrea Fusiello. For uncalibrated rectification, see Robert *et al.* [20].

The eight-point algorithm is due to Longuet-Higgins [11]; the normalization procedure to avoid numerical instabilities (discussed at length in Exercise 7.6) is due to Hartley [7]. Linear and nonlinear methods for determining the fundamental matrix, as well as stability issues and critical configurations, are studied by Luong and Faugeras [13]. Shashua [21] proposed a method for locating the epipoles and achieving projective reconstruction that require only six point matches. The recovery of the extrinsic parameters from the essential matrix described in this chapter is again due to Longuet-Higgins [11]. An alternative method for the calibration of the extrinsic parameters can be found in [8]. We have largely based the introduction to uncalibrated stereo on the seminal paper by Faugeras [4]. Similar results are discussed by Sparr [22] and Mohr and Arbogast [18]. More recently a number of methods for the recovery of Euclidean structure from the projective reconstruction have been proposed (see [3, 6, 19] for example).

If you are curious about understanding and creating autostereograms, check out the Web site `http://www.ccc.nottingham.ac.uk/~etzpc/sirds.html`. The face reconstruction in Figure 7.2 was computed by a stereo system commercialized by the Turing Institute (`http://www.turing.gla.ac.uk`), which maintains an interesting Web site on stereopsis. The INRIA-Syntim Web site contains useful test data, including calibrated stereo pairs (please notice the copyright attached!): `http://www-syntim.inria.fr/syntim/analyze/paires-eng.html`.

To conclude, we observe that stereo-like visual systems can also be built taking two pictures of the same scene from the same viewpoint, but under different illuminations. This is the so-called *photometric stereo*, first proposed by Woodham [23].

7.7 Review

Questions

- ❒ *7.1* What are the intrinsic and extrinsic parameters of a stereo system?
- ❒ *7.2* What are the main properties of correlation-based and feature-based methods for finding correspondences?
- ❒ *7.3* What is the epipolar constraint and how could you use it to speed up the search for corresponding points?
- ❒ *7.4* What are the main properties of the essential and fundamental matrices?
- ❒ *7.5* What is the purpose of rectification?
- ❒ *7.6* What happens in rectification if the focal lengths of the two original cameras are not equal?
- ❒ *7.7* What sort of 3-D reconstruction can be obtained if all the parameters, only the intrinsic parameters, or no parameters can be assumed to be known?
- ❒ *7.8* What is the purpose of step 4 in algorithm EUCLID_REC?
- ❒ *7.9* Is the reconstruction of the fundamental matrix necessary for uncalibrated stereo?
- ❒ *7.10* How can (7.23) reconstruct depths in millimeters if the focal length in millimeters is not known?

Exercises

- ○ *7.1* Estimate the accuracy of the simple stereo system of Figure 7.4 assuming that the only source of noise is the localization of corresponding points in the two images. (*Hint:* Take the partial derivatives of Z with respect to x, T, f.) Discuss the dependence of the error in depth estimation as a function of the baseline width and the focal length.
- ○ *7.2* Using your solution to Exercise 7.1, estimate the accuracy with which features should be localized in the two images in order to reconstruct depth with a relative error smaller than 1%.
- ○ *7.3* Check what happens if you compute SSD and cross-correlation between an arbitrary pattern and a perfectly black pattern over a window W. Discuss the effect of replacing the definition of cross-correlation with the *normalized cross-correlation*

$$\psi(x, y) = \frac{(x - \bar{x})(y - \bar{y})}{N_x N_y} \tag{7.46}$$

where $\bar{x} = \sum_W x$, $\bar{y} = \sum_W y$, $N_x = \sqrt{\sum_W x^2}$, and $N_y = \sqrt{\sum_W y^2}$. Can you precompute and store the possible values of ψ if you are using (7.46)?

- ○ *7.4* Discuss strategies for estimating correspondences at subpixel precision using correlation-based methods. (*Hint:* Keep track of the values of ψ in the neighborhood of the maximum, and take a weighted average of every integer location in that neighborhood.) Make sure that the weights are positive and sum to 1.
- ○ *7.5* Design a correlation-based method that can be used to match edge points.
- ○ *7.6* Determine the matrices H_l and H_r needed to normalize the entries of the fundamental matrix before applying algorithm EIGHT_POINT. (*Hint:* Given a set of points $\mathbf{p}_i = [x_i, y_i, 1]^\top$ with $i = 1, \ldots, n$, define $\bar{x} = \sum_i x_i/n$, $\bar{y} = \sum_i y_i/n$, and

$$\bar{d} = \frac{\sum_i \sqrt{(x_i - \bar{x})^2 + (y_i - \bar{y})^2}}{n\sqrt{2}}.$$

Then, find the 3×3 matrix H such that

$$H\mathbf{p}_i = \hat{\mathbf{p}}_i$$

with $\mathbf{p}_i = [(x_i - \bar{x})/d, (y_i - \bar{y})/d, 1]^\top$.) Verify that the average length of each component of $\hat{\mathbf{p}}_i$ equals 1.

- ○ *7.7* Write an algorithm reconstructing a scene from a rectified stereo pair using rectified images coordinates. (*Hint:* Use the simultaneous projection equations associated to a 3-D point in the two cameras.)
- ○ *7.8* Verify that (7.2) can be derived from (7.23) in the special case of the stereo system of Figure 7.4.
- ○ *7.9* In analogy with the case of point matching, compute the solution to the triangulation problem in the case of line matching. If l_l and l_r are the matched lines, this amounts to find the 3-D line intersection of the planes through O_l and l_l, and O_r and l_r respectively. Why is the triangulation based on lines computationally easier than the triangulation based on points?
- ○ *7.10* Assume $\mathbf{T}_l$ and R_l, and $\mathbf{T}_r$ and R_r are the extrinsic parameters of two cameras with respect to the same world reference frame. Show that the translation vector, $\mathbf{T}$, and rotation matrix, R, which define the extrinsic parameters of the stereo system composed of the two cameras are given by (7.24). (*Hint:* For a point $\mathbf{P}$ in the world reference frame we have $\mathbf{P}_r = R_r\mathbf{P} + \mathbf{T}_r$ and $\mathbf{P}_l = R_l\mathbf{P} + \mathbf{T}_l$. But the relation between $\mathbf{P}_l$ and $\mathbf{P}_r$ is given by $\mathbf{P}_r = R(\mathbf{P}_l - \mathbf{T})$.)
- ○ *7.11* In the same notation of Section 7.4, let $\mathbf{w}_i = \hat{\mathbf{E}}_i \times \hat{\mathbf{T}}_i$ with $\hat{\mathbf{E}}_i = \hat{\mathbf{T}}_i \times \mathbf{R}_i$. Prove that

$$(\hat{\mathbf{T}}^\top \mathbf{R}_i)\hat{\mathbf{T}} = \mathbf{w}_j \times \mathbf{w}_k$$

for every triplet (i, j, k) which is a cyclic permutation of $(1, 2, 3)$ and use the result to derive (7.28). (*Hint:* Make use of the vector identity $\mathbf{A} \times (\mathbf{B} \times \mathbf{C}) = (\mathbf{A}^\top\mathbf{C})\mathbf{B} - (\mathbf{A}^\top\mathbf{B})\mathbf{C}$.)

Projects

- *7.1* In a typical implementation of CORR-MATCHING, one computes $c(d) = \sum_W \psi$ at each pixel for each possible shift (where W is the window over which $c(d)$ is evaluated and ψ is the pixelwise cross-correlation measure), and stores the shift for which $c(d)$ is maximum. If the size of W is $n \times n$, this implementation requires $O(n^2)$ additions. However, if n is larger than a few units, the overlap between the correlation windows centered at neighbor pixels can be exploited to obtain a more efficient implementation that requires $O(2n)$ additions. The key idea is to compute $c(d)$ for each possible shift at all pixels first. This makes it possible to use the result of the computation of $c(d)$ for some d at one pixel to evaluate $c(d)$ at the neighboring pixel. Here is a simple way to do it. For each possible shift, evaluate ψ over the entire image. Once you have obtained $c(d)$ at some pixel p over the window W, you can compute $c(d)$ for the pixel immediately to the right of p, for example, by simply subtracting the contribution to $c(d)$ from the leftmost column of W and adding the contribution from the column immediately to the right of W. The memory requirement is not much different, as for each shift you do not need to save the value of ψ over the entire image, but only the intermediate maximum of $c(d)$ (and corresponding shift) for all pixels. Implement this version of CORR-MATCHING and compare it with the standard implementation.
- *7.2* Design and implement a program that, given a stereo pair, determines at least eight point matches, then recovers the fundamental matrix and the location of the epipoles. Check the accuracy of the result by measuring the distance between the estimated epipolar lines and image points not used by the matrix estimation.

References

[1] N. Ayache, *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*, MIT Press, Cambridge (MA) (1991).

[2] N. Ayache and B. Faverjon, Efficient Registration of Stereo Images by Matching Graph Descriptions of Edge Segments, *International Journal of Computer Vision*, Vol. 1, no. 2 (1987).

[3] F. Devernay and O.D. Faugeras, From Projective to Euclidean Reconstruction, *Technical Report 2725*, INRIA (1995) (available from `http://www.inria.fr`).

[4] O.D. Faugeras, What Can Be Seen in Three Dimensions with an Uncalibrated Stereo Rig?, *Proc. 2nd European Conference on Computer Vision*, Santa Margherita (Italy), pp. 563-578 (1992).

[5] O.D. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge (MA) (1993).

[6] R.I. Hartley, Estimation of Relative Camera Positions for Uncalibrated Cameras, *Proc. 2nd European Conference on Computer Vision*, Santa Margherita (Italy), pp. 579-587 (1992).

[7] R.I. Hartley, In Defence of the 8-Point Algorithm, *Proc. 5th International Conference on Computer Vision*, Cambridge (MA), pp. 1064-1070 (1995).

[8] B.K.P. Horn, Relative Orientation, *International Journal of Computer Vision*, Vol. 4, pp. 59-78 (1990).

[9] D.G. Jones and J. Malik, Determining 3-D Shape from Orientation and Spatial Frequency Disparities, *Proc. 2nd European Conference on Computer Vision*, Santa Margherita (Italy), pp. 661-669 (1992).

[10] T. Kanade and M. Okutomi, A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiments, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-16, pp. 920-932 (1994).

[11] H.C. Longuet-Higgins, A Computer Algorithm for Reconstructing a Scene from Two Projections, *Nature*, Vol. 293, no. 10, pp. 133-135 (1981).

[12] B.D. Lucas and T. Kanade, An Iterative Image Registration Technique with an Application to Stereo Vision, *Proc. International Joint Conference on Artificial Intelligence*, pp. 674-679 (1981).

[13] Q.-T. Luong and O.D. Faugeras, The Fundamental Matrix: Theory, Algorithms, and Stability Analysis, *International Journal of Computer Vision*, Vol. 17, pp. 43-75 (1996).

[14] D. Marr and T. Poggio, Cooperative Computation of Stereo Disparity, *Science* Vol. 194, pp. 283-287 (1976).

[15] D. Marr and T. Poggio, A Computational Theory of Human Stereo Vision, *Proc. R. Soc. Lond. B* Vol. 204, pp. 301-328 (1979).

[16] L. Matthies, T. Kanade and R. Szeliski, Kalman Filter-Based Algorithms for Estimating Depth from Image Sequences, *International Journal of Computer Vision*, Vol. 3, pp. 209-236 (1989).

[17] J.E.W. Mayhew and J.P. Frisby, Psychophysical and Computational Studies Towards a Theory of Human Stereopsis, *Artificial Intelligence*, Vol. 17, pp. 349-385 (1981).

[18] R. Mohr and E. Arbogast, It Can Be Done without Camera Calibration, *Pattern Recognition Letters*, Vol. 12, pp. 39-43 (1990).

[19] M. Pollefeys, L. Van Gool and M. Proesmans, Euclidean 3-D Reconstruction from Image Sequences with Variable Focal Lenghts, *Proc. European Conference on Computer Vision*, Cambridge (UK), pp. 31-42 (1996).

[20] L. Robert, C. Zeller, O.D. Faugeras and M. Hebert, Applications of Non-Metric Vision to Some Visually Guided Robotic Tasks, *Technical Report 2584*, INRIA (1995) (available from `http://www.inria.fr`).

[21] A. Shashua, Projective Depth: a Geometric Invariant for 3-D Reconstruction from Two Perspective/Orthographic Views and for Visual Recognition, *Proc. IEEE Int. Conf. on Computer Vision*, Berlin (Germany), pp. 583-590 (1993).

[22] G. Sparr, An Algebraic-Analytic Method for Reconstruction from Image Correspondences, in *Proc. 7th Scandinavian Conference on Image Analysis*, pp. 274-281 (1991).

[23] R. J. Woodham, Photometric Stereo: a Reflectance Map Technique for Determining Surface Orientation from a Single View, *Proc. SPIE Technical Symposium on Image Understanding Systems and Industrial Applications*, Vol. 155, pp. 136-143 (1978).