Multiple motion hypotheses

In the case of multiple moving objects in the scene, image points may no longer satisfy the same epipolar constraint. For example, if we know that there are two independent moving objects with motions, say (R^1, T^1) and (R^2, T^2), then the two images $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ of a point p on one of these objects should satisfy instead the equation

$$(\boldsymbol{x}_2^T E^1 \boldsymbol{x}_1)(\boldsymbol{x}_2^T E^2 \boldsymbol{x}_1) = 0, \tag{5.16}$$

corresponding to the fact that the point p moves according to either motion 1 or motion 2. Here $E^1 = \widehat{T^1} R^1$ and $E^2 = \widehat{T^2} R^2$. As we will see, from this equation it is still possible to recover E^1 and E^2 if enough points are visible on either object. Generalizing to more than two independent motions requires some attention; we will study the multiple-motion problem in Chapter 7.

5.2.2 Euclidean constraints and structure reconstruction

The eight-point algorithm just described uses as input a set of eight or more point correspondences and returns the relative pose (rotation and translation) between the two cameras up to an arbitrary scale $\gamma \in \mathbb{R}^+$. Without loss of generality, we may assume this scale to be $\gamma = 1$, which is equivalent to scaling translation to unit length. Relative pose and point correspondences can then be used to retrieve the position of the points in 3-D by recovering their depths relative to each camera frame.

Consider the basic rigid-body equation, where the pose (R, T) has been recovered, with the translation T defined up to the scale γ. In terms of the images and the depths, it is given by

$$\lambda_2^j \boldsymbol{x}_2^j = \lambda_1^j R \boldsymbol{x}_1^j + \gamma T, \quad j = 1, 2, \ldots, n. \tag{5.17}$$

Notice that since (R, T) are known, the equations given by (5.17) are linear in both the structural scale λ's and the motion scale γ's, and therefore they can be easily solved. For each point, λ_1, λ_2 are its depths with respect to the first and second camera frames, respectively. One of them is therefore redundant; for instance, if λ_1 is known, λ_2 is simply a function of (R, T). Hence we can eliminate, say, λ_2 from the above equation by multiplying both sides by $\widehat{\boldsymbol{x}_2}$, which yields

$$\lambda_1^j \widehat{\boldsymbol{x}_2^j} R \boldsymbol{x}_1^j + \gamma \widehat{\boldsymbol{x}_2^j} T = 0, \quad j = 1, 2, \ldots, n. \tag{5.18}$$

This is equivalent to solving the linear equation

$$M^j \bar{\lambda}^j \doteq \left[\widehat{\boldsymbol{x}_2^j} R \boldsymbol{x}_1^j, \ \widehat{\boldsymbol{x}_2^j} T \right] \begin{bmatrix} \lambda_1^j \\ \gamma \end{bmatrix} = 0, \tag{5.19}$$

where $M^j = \left[\widehat{\boldsymbol{x}_2^j} R \boldsymbol{x}_1^j, \ \widehat{\boldsymbol{x}_2^j} T \right] \in \mathbb{R}^{3 \times 2}$ and $\bar{\lambda}^j = [\lambda_1^j, \gamma]^T \in \mathbb{R}^2$, for $j = 1, 2, \ldots, n$. In order to have a unique solution, the matrix M^j needs to be of

rank 1. This is not the case only when $\widehat{\boldsymbol{x}_2} T = 0$, i.e. when the point p lies on the line connecting the two optical centers o_1 and o_2.

Notice that all the n equations above share the same γ; we define a vector $\vec{\lambda} = [\lambda_1^1, \lambda_1^2, \ldots, \lambda_1^n, \gamma]^T \in \mathbb{R}^{n+1}$ and a matrix $M \in \mathbb{R}^{3n \times (n+1)}$ as

$$M \doteq \begin{bmatrix} \widehat{\boldsymbol{x}_2^1} R \boldsymbol{x}_1^1 & 0 & 0 & 0 & 0 & \widehat{\boldsymbol{x}_2^1} T \\ 0 & \widehat{\boldsymbol{x}_2^2} R \boldsymbol{x}_1^2 & 0 & 0 & 0 & \widehat{\boldsymbol{x}_2^2} T \\ 0 & 0 & \ddots & 0 & 0 & \vdots \\ 0 & 0 & 0 & \widehat{\boldsymbol{x}_2^{n-1}} R \boldsymbol{x}_1^{n-1} & 0 & \widehat{\boldsymbol{x}_2^{n-1}} T \\ 0 & 0 & 0 & 0 & \widehat{\boldsymbol{x}_2^n} R \boldsymbol{x}_1^n & \widehat{\boldsymbol{x}_2^n} T \end{bmatrix}. \quad (5.20)$$

Then the equation

$$M \vec{\lambda} = 0 \quad (5.21)$$

determines all the unknown depths *up to a single universal scale*. The linear least-squares estimate of $\vec{\lambda}$ is simply the eigenvector of $M^T M$ that corresponds to its smallest eigenvalue. Note that this scale ambiguity is intrinsic, since without any prior knowledge about the scene and camera motion, one cannot disambiguate whether the camera moved twice the distance while looking at a scene twice larger but two times further away.

5.2.3 *Optimal pose and structure*

The eight-point algorithm given in the previous section assumes that *exact* point correspondences are given. In the presence of noise in image correspondences, we have suggested possible ways of estimating the essential matrix by solving a least-squares problem followed by a projection onto the essential space. But in practice, this will not be satisfying in at least two respects:

1. There is no guarantee that the estimated pose (R, T), is as close as possible to the true solution.
2. Even if we were to accept such an (R, T), a noisy image pair, say $(\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2)$, would not necessarily give rise to a consistent 3-D reconstruction, as shown in Figure 5.6.

At this stage of development, we do not want to bring in all the technical details associated with optimal estimation, since they would bury the geometric intuition. We will therefore discuss only the key ideas, and leave the technical details to Appendix 5.A as well as Chapter 11, where we will address more practical issues.

Choice of optimization objectives

Recall from Chapter 3 that a calibrated camera can be described as a plane perpendicular to the z-axis at a distance 1 from the origin; therefore, the coordinates of image points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are of the form $[x, y, 1]^T \in \mathbb{R}^3$. In practice, we cannot