PART 2

Basic Methods in Computer Vision and Image Processing

Chapter 4

LATEST ALGORITHMS FOR 3-D RECONSTRUCTION FROM TWO VIEWS

Kenichi Kanatani

Department of Computer Science, Okayama University, Okayama 700-8530 Japan
E-mail: kanatani@suri.cs.okayama-u.ac.jp

Yasuyuki Sugaya, and Yasushi Kanazawa

Department of Computer Science and Engineering, Toyohashi University of Technology, Toyohashi, Aichi 441-8580 Japan
E-mail: sugaya@iim.cs.tut.ac.jp kanazawa@cs.tut.ac.jp

4.1 Introduction

The problem of computing from point correspondences over two views their 3-D locations, the positions and orientations of the two cameras, and their intrinsic parameters was intensively studied in 1980s and 1990s. Today, it is a well established technology and is used for many applications including robot navigation and virtual reality as a typical example of what is known as *structure from motion* [Kanatani (1996); Hartley and Zisserman (2000)]. The principle is now extended to multiple views and video streams [Hartley and Zisserman (2000); Kanatani (2008)].

Although the mathematical principle is well understood today, ceaseless efforts have still been made to improve component technologies, such as the fundamental matrix computation, with a view to further upgrade the accuracy and computational efficiency. This chapter describes such latest efforts for two view reconstruction. The basic structure of the algorithm was given by Kanatani and Ohta (2003), and we replace its individual components by latest results. Specifically:

(1) Given point correspondences, we must first compute the fundamental matrix, for which Kanatani and Ohta (2003) used a method called *renormalization* [Kanatani (2000)] and then optimally corrected the resulting solution so that it satisfies the rank constraint. This produces a statistically optimal solution up to high order noise terms. Here, we compute the fundamental matrix by a strictly optimal scheme recently introduced by Kanatani and Sugaya (2009), using the *EFNS* (*Extended Fundamental Numerical Scheme*) [Kanatani and

Sugaya (2007b)]. This not only achieves yet higher accuracy but also significantly simplifies the algorithm.

(2) Optimal estimation of the fundamental matrix requires iterations. Kanatani and Ohta (2003) initialized their renormalization by least squares. Here, we use the method of Taubin (1991), which is known to produce a better solution without any iterations [Kanatani and Sugaya (2007a,c)]. This considerably accelerates the convergence of the succeeding optimization of the fundamental matrix.

(3) From the computed fundamental matrix, we next compute the focal lengths of the two cameras. For this, Kanatani and Ohta (2003) used the formula of Bougnoux (1998), which we call the "free focal length method". This method fails when the two camera are in a degenerate configuration [Kanatani et al. (2006)]. However, this configuration occurs very frequently in practical situations. Following Kanatani and Matsunaga (2000) and Kanatani et al. (2006), we also describe the "fixed focal length method", assuming that the two focal lengths are equal.

(4) After the focal lengths are computed, we can determine the relative camera positions [Kanatani (1993)], from which we can triangulate the 3-D locations of the points in the scene. For this, we use the method of Kanatani et al. (2008) for optimally correcting corresponding points and triangulating them relative to a given world coordinate system.

This chapter is organized as follows. Section 4.2 describes the entire structure of our algorithm. Section 4.3 focuses on the fundamental matrix computation. We describe the details of the initialization by the Taubin method and the subsequent optimization using the EFNS procedure. In Sec. 4.4, we discuss in detail the computation of the camera focal lengths. We describe the general theory, the free focal length method, and the fixed focal length method. Section 4.5 concentrates on computation of 3-D motion and structure from the computed focal lengths and the fundamental matrix. We describe practical procedures for computing the camera translation and rotation and optimally triangulating individual points. We conclude in Sec. 4.6, summarizing related issues and listing typical situations in which 3-D reconstruction fails if care is not taken. The Appendix describes the details of computational procedures omitted in the body.

4.2 Entire Algorithm Structure

Figure 4.1 shows the entire structure of the algorithm to be described here. The input and output are as follows:

Input: The image coordinates (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$ (≥ 8), of corresponding points.

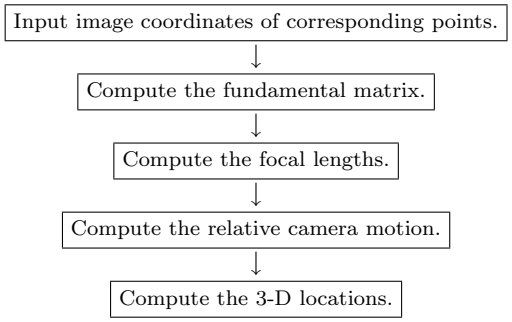


Fig. 4.1 The entire structure of our algorithm.

Output: Their 3-D locations $(X_\alpha, Y_\alpha, Z_\alpha)$, $\alpha = 1, ..., N$.

Our algorithm is based on the following assumptions and conventions:

- The principal point (the intersection of the optical axis and the image plane) is known. We define an xy image coordinate system with the origin $(0, 0)$ at the principal point. We take the x-axis *upward* direction and the y-axis *rightward*.
- Image distortion does not exists or already corrected beforehand. Hence, the rows and columns of the pixel array are regarded as physically orthogonal with aspect ratio 1.
- The focal lengths of the two cameras are equal, or equivalently one camera is moved without changing the zoom or the focus.

Given sufficiently many point correspondences over sufficiently many images, we can in principle determine the principal point and also correct the image distortion [Hartley and Zisserman (2000)]. From two views, however, all available information is the epipolar constraint encoded in the 3×3 fundamental matrix (Sec. 4.3.1). The fundamental matrix has scale indeterminacy, so only five of its elements are independent. From this limited information, 3-D reconstruction is possible only when the principal point is known and no image distortion exists (Sec. 4.4.1). For most of the digital cameras available on the market today, we can safely assume this. Otherwise, we need prior camera calibration and geometric correction.

The reason we take the x-axis upward and the y-axis rightward is that we view the camera optical axis, which we identify with the z-axis, as extending *away* from the viewer. For humans, this is a natural interpretation of perspective projection. Taking the x-axis rightward and the y-axis upward would define a *left-handed* coordinate system, in which rotation matrices have determinant -1. To avoid this, we could of course take the x-axis downward and the y-axis rightward.

Thus, we model the camera imaging geometry as shown in Fig. 4.2. We take an XYZ coordinate system fixed to the camera in such a way that the origin O is at the center of the lens, which we hereafter call the *viewpoint*, and the Z axis

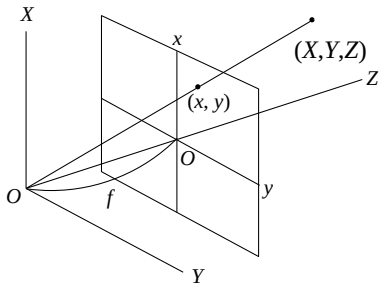


Fig. 4.2 Interpretation of perspective projection.

along the optical axis of the lens . We identify the xy image plane with the plane $Z = f$, and call f the *focal length*. The xy image coordinate system is defined so that the image origin $(0, 0)$ is at the *principal point* $(0, 0, f)$ and the x- and y-axis are parallel to the X- and Y-axis, respectively. A point (X, Y, Z) in the scene is projected to the intersection (x, y) of the image plane with the ray, or the *line of sight*, starting from the viewpoint O and passing through (X, Y, Z). This geometry is known as *perspective projection* or the *pin-hole camera model*.

The assumption of equal focal lengths is not necessary as long as the camera configuration is nondegenerate. However, most practical applications take place in a degenerate configuration, in which 3-D reconstruction is impossible unless the two cameras have the same focal length or their focal lengths are known (Sec. 4.4.3). The equal focal length assumption is not a strong constraint, since people usually take images without changing the zoom or the focus. The subsequent sections describe the component algorithms according to the flow shown in Fig. 4.1.

4.3 Fundamental Matrix Computation

4.3.1 *Epipolar equation*

The first step is to compute the fundamental matrix $\boldsymbol{F}$ from point correspondences (x_α, y_α) and (x'_α, y'_α), $\alpha = 1, ..., N$. We regard x_α, y_α, x'_α, and y'_α as noisy observations of their true values $\bar{x}_\alpha$, $\bar{y}_\alpha$, $\bar{x}'_\alpha$, and $\bar{y}'_\alpha$. We represent points (x_α, y_α) and (x'_α, y'_α) by 3-D vectors

$$\boldsymbol{x}_\alpha = \begin{pmatrix} x_\alpha/f_0 \\ y_\alpha/f_0 \\ 1 \end{pmatrix}, \qquad \boldsymbol{x}'_\alpha = \begin{pmatrix} x'_\alpha/f_0 \\ y'_\alpha/f_0 \\ 1 \end{pmatrix}, \tag{4.1}$$

where f_0 is a scaling constant of the order of the image size for stabilizing numerical computation [Hartley (1997)]. Let $\bar{\boldsymbol{x}}_\alpha$ and $\bar{\boldsymbol{x}}'_\alpha$ be the true values of $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$, respectively. They satisfy the following *epipolar equation* [Hartley and Zisserman (2000)]:

$$(\bar{\boldsymbol{x}}_\alpha, \boldsymbol{F}\bar{\boldsymbol{x}}'_\alpha) = 0. \tag{4.2}$$

Throughout this chapter, we denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $(\boldsymbol{a}, \boldsymbol{b})$. The matrix $\boldsymbol{F}$ in Eq. (4.2) is of rank 2 and called the *fundamental matrix*; it depends on the relative positions and orientations of the two cameras and their intrinsic parameters (e.g., their focal lengths) but not on the scene or the choice of the corresponding points. As Eq. (4.2) implies, the matrix $\boldsymbol{F}$ has scale indeterminacy.

If the noise in (x_α, y_α) and (x'_α, y'_α) is regarded as independent and identical Gaussian variables of mean 0, *maximum likelihood* (*ML*) estimators of $\bar{\boldsymbol{x}}_\alpha$, $\bar{\boldsymbol{x}}'_\alpha$, and $\boldsymbol{F}$ are computed by minimizing the *reprojection error*

$$E = \sum_{\alpha=1}^{N} \Big(\|\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha\|^2 + \|\boldsymbol{x}'_\alpha - \bar{\boldsymbol{x}}'_\alpha\|^2 \Big), \tag{4.3}$$

subject to Eq. (4.2). In order to determine the nine elements of $\boldsymbol{F}$ up to scale from Eq. (4.2), we need to observe at least eight points. If we consider the rank constraint $\det \boldsymbol{F} = 0$, we could theoretically determine $\boldsymbol{F}$ from seven correspondences, but in that case multiple solutions can exist [Hartley and Zisserman (2000)]. Hereafter, we assume that we observe eight or more points.

If the points we observe are in a special configuration in 3-D, however, Eq. (4.2) can be satisfied by infinitely many $\boldsymbol{F}$. This occurs if all the points and the viewpoints O and O' of the two cameras are on a quadratic surface, known as a *critical surface* [Maybank (1990, 1991), Kanatani (1993, 1996)]. Such a configuration occurs, e.g., when all the points are coplanar in the scene. In the following, we assume that the points are in general position so that Eq. (4.2) is satisfied by a unique $\boldsymbol{F}$ up to scale. We will return to this issue in Sec. 4.6.

4.3.2 *Initializing the fundamental matrix*

All known algorithms for minimizing Eq. (4.3) subject to Eq. (4.2) are iterative, so we need an appropriate initial value. The simplest and most popular is the least squares (Appendix A). As pointed out by Kanatani and Sugaya (2007a,c), however, the accuracy of the least squares is rather limited, and a much better solution is obtained by the method of Taubin (1991). The use of the Taubin solution as the initial value considerably accelerates the convergence of the subsequent optimization iterations. For computational convenience, we identify the fundamental matrix $\boldsymbol{F}$ with the 9-D vector

$$\boldsymbol{u} = (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})^\top. \tag{4.4}$$

The Taubin method goes as follows (see Appendix B for the derivation):

Input: The image coordinates (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$ (≥ 8), of corresponding points.

Output: The 9-D vector encoding the fundamental matrix $\boldsymbol{F}$.

(1) Represent the corresponding points (x_α, y_α) and (x'_α, y'_α) by the following 8-D vectors:

$$\boldsymbol{z}_\alpha = (x_\alpha x'_\alpha,\ x_\alpha y'_\alpha,\ f_0 x_\alpha,\ y_\alpha x'_\alpha,\ y_\alpha y'_\alpha,\ f_0 y_\alpha,\ f_0 x'_\alpha,\ f_0 y'_\alpha)^\top. \tag{4.5}$$

(2) Compute the following 8×8 matrices $V_0[\boldsymbol{z}_\alpha]$:

$$V_0[\boldsymbol{z}_\alpha] = \begin{pmatrix} x_\alpha^2 + x_\alpha'^2 & x'_\alpha y'_\alpha & f_0 x'_\alpha & x_\alpha y_\alpha & 0 & 0 & f_0 x_\alpha & 0 \\ x'_\alpha y'_\alpha & x_\alpha^2 + y_\alpha'^2 & f_0 y'_\alpha & 0 & x_\alpha y_\alpha & 0 & 0 & f_0 x_\alpha \\ f_0 x'_\alpha & f_0 y'_\alpha & f_0^2 & 0 & 0 & 0 & 0 & 0 \\ x_\alpha y_\alpha & 0 & 0 & y_\alpha^2 + x_\alpha'^2 & x'_\alpha y'_\alpha & f_0 x'_\alpha & f_0 y_\alpha & 0 \\ 0 & x_\alpha y_\alpha & 0 & x'_\alpha y'_\alpha & y_\alpha^2 + y_\alpha'^2 & f_0 y'_\alpha & 0 & f_0 y_\alpha \\ 0 & 0 & 0 & f_0 x'_\alpha & f_0 y'_\alpha & f_0^2 & 0 & 0 \\ f_0 x_\alpha & 0 & 0 & f_0 y_\alpha & 0 & 0 & f_0^2 & 0 \\ 0 & f_0 x_\alpha & 0 & 0 & f_0 y_\alpha & 0 & 0 & f_0^2 \end{pmatrix}. \tag{4.6}$$

(3) Compute the following 8-D vectors $\bar{\boldsymbol{z}}$ and $\tilde{\boldsymbol{z}}_\alpha$:

$$\bar{\boldsymbol{z}} = \frac{1}{N}\sum_{\alpha=1}^{N} \boldsymbol{z}_\alpha, \qquad \tilde{\boldsymbol{z}}_\alpha = \boldsymbol{z}_\alpha - \bar{\boldsymbol{z}}. \tag{4.7}$$

(4) Compute the following 8×8 matrices $\boldsymbol{M}_{\mathrm{TB}}$ and $\boldsymbol{N}_{\mathrm{TB}}$:

$$\boldsymbol{M}_{\mathrm{TB}} = \sum_{\alpha=1}^{N} \tilde{\boldsymbol{z}}_\alpha \tilde{\boldsymbol{z}}_\alpha^\top, \qquad \boldsymbol{N}_{\mathrm{TB}} = \sum_{\alpha=1}^{N} V_0[\boldsymbol{z}_\alpha]. \tag{4.8}$$

(5) Solve the following generalized eigenvalue problem (Appendix C) and compute the unit generalized eigenvector $\boldsymbol{v}$ for the smallest generalized eigenvalue:

$$\boldsymbol{M}_{\mathrm{TB}}\boldsymbol{v} = \lambda \boldsymbol{N}_{\mathrm{TB}}\boldsymbol{v}. \tag{4.9}$$

(6) Return the 9-D unit vector $\boldsymbol{u}$

$$\boldsymbol{u} = \mathcal{N}[\begin{pmatrix} \boldsymbol{v} \\ -(\bar{\boldsymbol{z}}, \boldsymbol{v})/f_0^2 \end{pmatrix}], \tag{4.10}$$

where $\mathcal{N}[\,\cdot\,]$ denotes normalization to unit norm ($\mathcal{N}[\boldsymbol{a}] = \boldsymbol{a}/\|\boldsymbol{a}\|$).

The use of f_0 in Eqs. (4.5) and (4.6) is for making the vector components and the matrix elements to have the same order of magnitude for stabilizing the numerical computation [Hartley (1997)]. The final solution does not depend on the specific value f_0.

4.3.3 *Alternating optimization scheme*

Starting from the Taubin solution, we iteratively optimize it so that it minimizes the reprojection error E in Eq. (4.3) subject to the epipolar equation in Eq. (4.2). The major obstacle is that the fundamental matrix $\boldsymbol{F}$ is constrained to have rank 2. A popular approach to this is to introduce auxiliary variables so that the rank constraint is automatically satisfied and to do numerical search in the augmented

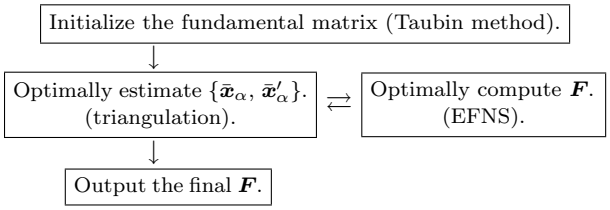


Fig. 4.3 Fundamental matrix computation.

parameter space. Specifically, we first compute from an assumed fundamental matrix a tentative 3-D reconstruction, i.e., the 3-D locations of the observed points, the relative positions and orientations of the two cameras, and their focal lengths. Then, we iteratively adjust the 3-D locations, the camera configuration, and the focal lengths so that the image positions obtained by "reprojecting" the reconstructed 3-D points are as close to the observed points as possible[1], see, e.g, Bartoli and Sturm (2004). This approach is known as *bundle adjustment* [Triggs et al. (2000)].

The search space for bundle adjustment is very high dimensional, since the N locations in 3-D are all taken to be variables as well as the camera configuration and the focal lengths. A well known optimization scheme is the Levernberg-Marquardt method [Press et al. (1992)], but the search is very inefficient unless one incorporates a clever implementation such as preprocessing of the Hessian, which is very high dimensional and very sparse. Here, we describe an efficient method recently proposed by Kanatani and Sugaya (2009). The computation is done only in the 9-D space of the fundamental matrix, yet is equivalent to bundle adjustment. The computation of $\{\bar{\boldsymbol{x}}_\alpha, \bar{\boldsymbol{x}}'_\alpha\}$, and $\boldsymbol{F}$ that minimize Eq. (4.3) subject to Eq. (4.2) consists of the following two components, which are alternated until the solution converges (Fig. 4.3):

- Optimally computing the estimate $\{\hat{\boldsymbol{x}}_\alpha, \hat{\boldsymbol{x}}'_\alpha\}$ of $\{\bar{\boldsymbol{x}}_\alpha, \bar{\boldsymbol{x}}'_\alpha\}$ for a given $\boldsymbol{F}$.
- Optimally computing $\boldsymbol{F}$ for a given estimate $\{\hat{\boldsymbol{x}}_\alpha, \hat{\boldsymbol{x}}'_\alpha\}$ of $\{\bar{\boldsymbol{x}}_\alpha, \bar{\boldsymbol{x}}'_\alpha\}$.

In the first component, we minimize Eq. (4.3) with respect to $\{\bar{\boldsymbol{x}}_\alpha, \bar{\boldsymbol{x}}'_\alpha\}$ subject to Eq. (4.2) for a given $\boldsymbol{F}$. This is nothing but what is known as *triangulation* of stereo images. Here, we use the procedure of Kanatani et al. (2008). In the second component, we minimize Eq. (4.3) with respect to $\boldsymbol{F}$ subject to Eq. (4.2) and the rank constraint $\det \boldsymbol{F} = 0$ for a given estimate of $\{\bar{\boldsymbol{x}}_\alpha, \bar{\boldsymbol{x}}'_\alpha\}$. For this computation, we use the *EFNS* (*Extended Fundamental Numerical Scheme*) of Kanatani and Sugaya (2007b). This scheme iteratively updates $\boldsymbol{F}$ so that the reprojection error E decreases and $\det \boldsymbol{F}$ approaches 0 at the same time. In other words, the rank constraint $\det \boldsymbol{F} = 0$ is not satisfied in the course of iterations but is satisfied in the end. This type of iterations was introduced by Chojnacki et al. (2004), who called their method *CFNS* (*Constrained Fundamental Numerical Scheme*), which is

[1]From this, Eq. (4.3) comes to be known as the "reprojection error".

a modification of their unconstrained optimization scheme called *FNS* (*Fundamental Numerical Scheme*) [Chojnacki et al. (2000)]. However, Kanatani and Sugaya (2007b) pointed out that the CFNS does not necessarily converge to a correct solution while their EFNS is guaranteed to produce an optimal solution.

4.3.4 *Optimal computation of the fundamental matrix*

Given an initial value of the 9-D vector representation $\boldsymbol{u}$ of the fundamental matrix $\boldsymbol{F}$ in the form of Eq. (4.4) and observed corresponding points (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$, the following algorithm optimally computes the fundamental matrix $\boldsymbol{F}$ (see Kanatani and Sugaya (2009) and Kanatani (2009) for the derivation).

Input: The image coordinates (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$ (≥ 8), of corresponding points, and the 9-D vector $\boldsymbol{u}$ representing an initial estimate of $\boldsymbol{F}$.
Output: Optimized fundamental matrix $\boldsymbol{F}$.

(1) Let $E^0 = \infty$ (a sufficiently large number), and initialize $\hat{x}_\alpha$, $\hat{y}_\alpha$, $\hat{x}'_\alpha$, $\hat{y}'_\alpha$, $\tilde{x}_\alpha$, $\tilde{y}_\alpha$, $\tilde{x}'_\alpha$, and $\tilde{y}'_\alpha$, $\alpha = 1, ..., N$, as follows:

$$\hat{x}_\alpha = x_\alpha, \quad \hat{y}_\alpha = y_\alpha, \quad \hat{x}'_\alpha = x'_\alpha, \quad \hat{y}'_\alpha = y'_\alpha,$$

$$\tilde{x}_\alpha = \tilde{y}_\alpha = \tilde{x}'_\alpha = \tilde{y}'_\alpha = 0. \tag{4.11}$$

(2) Compute the following 9-D vectors $\boldsymbol{\xi}_\alpha$ and 9×9 matrices $V_0[\boldsymbol{\xi}_\alpha]$, $\alpha = 1, ..., N$:

$$\boldsymbol{\xi}_\alpha = \begin{pmatrix} \hat{x}_\alpha\hat{x}'_\alpha + \hat{x}'_\alpha\tilde{x}_\alpha + \hat{x}_\alpha\tilde{x}'_\alpha \\ \hat{x}_\alpha\hat{y}'_\alpha + \hat{y}'_\alpha\tilde{x}_\alpha + \hat{x}_\alpha\tilde{y}'_\alpha \\ f_0(\hat{x}_\alpha + \tilde{x}_\alpha) \\ \hat{y}_\alpha\hat{x}'_\alpha + \hat{x}'_\alpha\tilde{y}_\alpha + \hat{y}_\alpha\tilde{x}'_\alpha \\ \hat{y}_\alpha\hat{y}'_\alpha + \hat{y}'_\alpha\tilde{y}_\alpha + \hat{y}_\alpha\tilde{y}'_\alpha \\ f_0(\hat{y}_\alpha + \tilde{y}_\alpha) \\ f_0(\hat{x}'_\alpha + \tilde{x}'_\alpha) \\ f_0(\hat{y}'_\alpha + \tilde{y}'_\alpha) \\ f_0^2 \end{pmatrix}, \tag{4.12}$$

$$V_0[\boldsymbol{\xi}_\alpha] = \begin{pmatrix} \hat{x}_\alpha^2 + \hat{x}_\alpha'^2 & \hat{x}'_\alpha\hat{y}'_\alpha & f_0\hat{x}'_\alpha & \hat{x}_\alpha\hat{y}_\alpha & 0 & 0 & f_0\hat{x}_\alpha & 0 & 0 \\ \hat{x}'_\alpha\hat{y}'_\alpha & \hat{x}_\alpha^2 + \hat{y}_\alpha'^2 & f_0\hat{y}'_\alpha & 0 & \hat{x}_\alpha\hat{y}_\alpha & 0 & 0 & f_0\hat{x}_\alpha & 0 \\ f_0\hat{x}'_\alpha & f_0\hat{y}'_\alpha & f_0^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hat{x}_\alpha\hat{y}_\alpha & 0 & 0 & \hat{y}_\alpha^2 + \hat{x}_\alpha'^2 & \hat{x}'_\alpha\hat{y}'_\alpha & f_0\hat{x}'_\alpha & f_0\hat{y}_\alpha & 0 & 0 \\ 0 & \hat{x}_\alpha\hat{y}_\alpha & 0 & \hat{x}'_\alpha\hat{y}'_\alpha & \hat{y}_\alpha^2 + \hat{y}_\alpha'^2 & f_0\hat{y}'_\alpha & 0 & f_0\hat{y}_\alpha & 0 \\ 0 & 0 & 0 & f_0\hat{x}'_\alpha & f_0\hat{y}'_\alpha & f_0^2 & 0 & 0 & 0 \\ f_0\hat{x}_\alpha & 0 & 0 & f_0\hat{y}_\alpha & 0 & 0 & f_0^2 & 0 & 0 \\ 0 & f_0\hat{x}_\alpha & 0 & 0 & f_0\hat{y}_\alpha & 0 & 0 & f_0^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.13}$$

(3) Update $\boldsymbol{u}$ by calling the subprogram EFNS in Sec. 4.3.5.
(4) Update $\tilde{x}_\alpha$, $\tilde{y}_\alpha$, $\tilde{x}'_\alpha$, and $\tilde{y}'_\alpha$, $\alpha = 1, ..., N$, as follows:

$$\begin{pmatrix} \tilde{x}_\alpha \\ \tilde{y}_\alpha \end{pmatrix} \leftarrow \frac{(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} \begin{pmatrix} u_1 & u_2 & u_3 \\ u_4 & u_5 & u_6 \end{pmatrix} \begin{pmatrix} \hat{x}'_\alpha \\ \hat{y}'_\alpha \\ f_0 \end{pmatrix},$$

$$\begin{pmatrix} \tilde{x}'_\alpha \\ \tilde{y}'_\alpha \end{pmatrix} \leftarrow \frac{(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} \begin{pmatrix} u_1 & u_4 & u_7 \\ u_2 & u_5 & u_8 \end{pmatrix} \begin{pmatrix} \hat{x}_\alpha \\ \hat{y}_\alpha \\ f_0 \end{pmatrix}. \tag{4.14}$$

(5) Update $\hat{x}_\alpha$, $\hat{y}_\alpha$, $\hat{x}'_\alpha$, and $\hat{y}'_\alpha$, $\alpha = 1, ..., N$, as follows:

$$\hat{x}_\alpha \leftarrow x_\alpha - \tilde{x}_\alpha, \quad \hat{y}_\alpha \leftarrow y_\alpha - \tilde{y}_\alpha,$$

$$\hat{x}'_\alpha \leftarrow x'_\alpha - \tilde{x}'_\alpha, \quad \hat{y}'_\alpha \leftarrow y'_\alpha - \tilde{y}'_\alpha. \tag{4.15}$$

(6) Compute the reprojection error E as follows:

$$E = \sum_{\alpha=1}^{N} (\tilde{x}_\alpha^2 + \tilde{y}_\alpha^2 + \tilde{x}'^{\,2}_\alpha + \tilde{y}'^{\,2}_\alpha). \tag{4.16}$$

(7) If $E \approx E^0$, output

$$\boldsymbol{F} = \begin{pmatrix} u_1 & u_2 & u_3 \\ u_4 & u_5 & u_6 \\ u_7 & u_8 & u_9 \end{pmatrix}. \tag{4.17}$$

Else, let $E^0 \leftarrow E$ and go back to Step (2).

4.3.5 *EFNS procedure*

The EFNS procedure called in the Step (3) of the above algorithm is the following computation [Kanatani and Sugaya (2007b)]. It produces the 9-D vector $\boldsymbol{u}$ that represents an optimal estimate of the fundamental matrix $\boldsymbol{F}$ for the current $(\hat{x}_\alpha, \hat{y}_\alpha)$, $(\hat{x}'_\alpha, \hat{y}'_\alpha)$, $\alpha = 1, ..., N$, that satisfies the rank constraint $\det \boldsymbol{F} = 0$.

Input: The 9-D vectors $\boldsymbol{\xi}_\alpha$, $\alpha = 1, ..., N$, that encode the corresponding points and the 9-D vector $\boldsymbol{u}$ that represents the current fundamental matrix.
Output: The 9-D vector $\boldsymbol{u}$ that represents the updated fundamental matrix.

(1) Compute the following 9×9 matrices $\boldsymbol{M}$ and $\boldsymbol{L}$:

$$\boldsymbol{M} = \sum_{\alpha=1}^{N} \frac{\boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}, \qquad \boldsymbol{L} = \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)^2 V_0[\boldsymbol{\xi}_\alpha]}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}. \tag{4.18}$$

(2) Compute the following 9-D vector $\boldsymbol{u}^\dagger$ and 9×9 matrix $\boldsymbol{P}_{\boldsymbol{u}^\dagger}$:

$$\boldsymbol{u}^\dagger = \mathcal{N}[\begin{pmatrix} u_5u_9 - u_8u_6 \\ u_6u_7 - u_9u_4 \\ u_4u_8 - u_7u_5 \\ u_8u_3 - u_2u_9 \\ u_9u_1 - u_3u_7 \\ u_7u_2 - u_1u_8 \\ u_2u_6 - u_5u_3 \\ u_3u_4 - u_6u_1 \\ u_1u_5 - u_4u_2 \end{pmatrix}], \qquad \boldsymbol{P}_{\boldsymbol{u}^\dagger} = \boldsymbol{I} - \boldsymbol{u}^\dagger\boldsymbol{u}^{\dagger\top}. \tag{4.19}$$

(3) Compute the following 9×9 matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$:

$$\boldsymbol{X} = \boldsymbol{M} - \boldsymbol{L}, \qquad \boldsymbol{Y} = \boldsymbol{P}_{\boldsymbol{u}^\dagger}\boldsymbol{X}\boldsymbol{P}_{\boldsymbol{u}^\dagger}. \tag{4.20}$$

(4) Compute the 9-D unit eigenvectors $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ for the smallest two eigenvalues of $\boldsymbol{Y}$.

(5) Compute the following 9-D vectors $\hat{\boldsymbol{u}}$ and $\boldsymbol{u}'$:

$$\hat{\boldsymbol{u}} = (\boldsymbol{u}, \boldsymbol{v}_1)\boldsymbol{v}_1 + (\boldsymbol{u}, \boldsymbol{v}_2)\boldsymbol{v}_2, \qquad \boldsymbol{u}' = \mathcal{N}[\boldsymbol{P}_{\boldsymbol{u}^\dagger}\hat{\boldsymbol{u}}]. \tag{4.21}$$

(6) If $\boldsymbol{u}' \approx \boldsymbol{u}$ up to sign, return $\boldsymbol{u}'$ as the update of $\boldsymbol{u}$. Else, let $\boldsymbol{u} \leftarrow \mathcal{N}[\boldsymbol{u} + \boldsymbol{u}']$ and go back to Step (1).

The vector $\boldsymbol{u}^\dagger$ in Eqs. (4.19) encodes the nine elements of the cofactor $\boldsymbol{F}^\dagger$ of the fundamental matrix $\boldsymbol{F}$. The matrix $\boldsymbol{P}_{\boldsymbol{u}^\dagger}$ represents projection of the 9-D space onto the 8-D subspace orthogonal to $\boldsymbol{u}^\dagger$. In Step (4), Kanatani and Sugaya (2009) computed two eigenvectors for the smallest eigenvalues *in absolute values*. Later, it has experimentally been found that simply computing two eigenvectors for the smallest eigenvalues results in better convergence in the presence of large noise. The same phenomenon was observed for the FNS of Chojnacki et al. (2000), for which computing the smallest eigenvalue, rather than the smallest eigenvalue in absolute value, exhibits better convergence [Kanatani and Sugaya (2007a,c)]. In Step (6), we could let $\boldsymbol{u} \leftarrow \boldsymbol{u}'$, but it has been observed that the value $\boldsymbol{u}'$ computed in Step (5) in the next round often reverts to the former value of $\boldsymbol{u}$, falling in infinite looping. To avoid this, the "midpoint" $(\boldsymbol{u}' + \boldsymbol{u})/2$ is normalized to a unit vector $\mathcal{N}[\boldsymbol{u}' + \boldsymbol{u}]$. This greatly improves convergence. In fact, it has been confirmed that this technique also improves the convergence of FNS of Chojnacki et al. (2000), which sometimes oscillates in the presence of very large noise.

4.4 Focal Length Computation

4.4.1 *General theory*

From thus computed fundamental matrix $\boldsymbol{F}$, we next compute the focal lengths f and f' (in pixels) of the two cameras and their relative translation $\boldsymbol{t}$ and rotation

$\boldsymbol{R}$, which we call the *motion parameters*. The fundamental matrix $\boldsymbol{F}$ has scale indeterminacy, and it should satisfy the rank constraint $\det \boldsymbol{F} = 0$, so it has seven degrees of freedom.

The scale indeterminacy of $\boldsymbol{F}$ corresponds to the scale indeterminacy of the translation $\boldsymbol{t}$, since we would observe the same images after the 3-D scene, the depths, and the camera translation $\boldsymbol{t}$ are simultaneously scaled up/down by the same ratio. In order to fix the scale, we normalize the translation $\boldsymbol{t}$ to unit norm ($\|\boldsymbol{t}\| = 1$), so it has two degrees of freedom. The rotation $\boldsymbol{R}$ has three degrees of freedom. Together with f and f', the unknowns have seven degrees of freedom. Thus, we should be able to compute the focal lengths $\{f,\ f'\}$ and the motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$ from $\boldsymbol{F}$.

If more views were observed, other camera parameters such as the aspect ratio, the skew angle, and the principal point (see Sec. 4.5.3) could be computed [Hartley and Zisserman (2000)]. As the above counting of the degrees of freedom indicates, however, we cannot reconstruct 3-D from two views unless other parameters are known. Hence, we assume an ideal camera for which the aspect ratio is 1 with no image skew, and the principal point is at the center of the image plane (Fig. 4.2). If the two cameras are ideal in the above sense, the fundamental matrix $\boldsymbol{F}$ has the following expression in the focal lengths $\{f,\ f'\}$ and the motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$ [Kanatani (1993), Hartley and Zisserman (2000)]:

$$\boldsymbol{F} = \mathrm{diag}(1, 1, \frac{f}{f_0})\, \boldsymbol{t} \times \boldsymbol{R}\, \mathrm{diag}(1, 1, \frac{f'}{f_0}). \tag{4.22}$$

The symbol $\mathrm{diag}(a, b, c)$ designates the diagonal matrix with diagonal elements a, b, and c in that order. The product $\boldsymbol{a} \times \boldsymbol{V}$ of a vector $\boldsymbol{a}$ and a matrix $\boldsymbol{V}$ is a matrix consisting of columns $\boldsymbol{a} \times \boldsymbol{v}_1$, $\boldsymbol{a} \times \boldsymbol{v}_2$, and $\boldsymbol{a} \times \boldsymbol{v}_3$, where $\boldsymbol{v}_i$ is the ith column of $\boldsymbol{V}$. We define the *essential matrix* $\boldsymbol{E}$ by

$$\boldsymbol{E} = \mathrm{diag}(1, 1, \frac{f_0}{f})\boldsymbol{F}\mathrm{diag}(1, 1, \frac{f_0}{f'})\, (= \boldsymbol{t} \times \boldsymbol{R}). \tag{4.23}$$

Huang and Faugeras (1988) proved that a matrix $\boldsymbol{E}$ can be written as $\boldsymbol{t} \times \boldsymbol{R}$ for some vector $\boldsymbol{t}$ and some rotation matrix $\boldsymbol{R}$ if and only if $\boldsymbol{E}$ has singular values σ, σ, and 0 for some $\sigma > 0$. Let us define the Frobenius norm of matrix $\boldsymbol{A}$ by $\|\boldsymbol{A}\| = \sqrt{\sum_{i,j=1}^{N} A_{ij}^2}$. Kanatani (1993) wrote this condition as

$$\|\boldsymbol{E}\boldsymbol{E}^\top\|^2 = \frac{1}{2}\|\boldsymbol{E}\|^4 \tag{4.24}$$

along with $\det \boldsymbol{E} = 0$, which is automatically satisfied by the rank constraint $\det \boldsymbol{F} = 0$ of $\boldsymbol{F}$. If $\sigma_1 \geq \sigma_2 \geq \sigma_3$ (≥ 0) are the singular values of $\boldsymbol{E}$, the rank constraint $\det \boldsymbol{E} = 0$ implies $\sigma_3 = 0$. It can easily be seen that Eq. (4.24) is then equivalent to $(\sigma_1^2 - \sigma_2^2)^2 = 0$ [Kanatani (1993)]. Thus, if Eq. (4.23) is substituted, Eq. (4.24) gives a set of equations for f and f'. Here, we describe the computational theory of Kanatani and Matsunaga (2000). First, we change the variables from f and f' to

$$\xi = \Big(\frac{f_0}{f}\Big)^2 - 1, \qquad \eta = \Big(\frac{f_0}{f'}\Big)^2 - 1. \tag{4.25}$$

Next, we define the function

$$K(\xi,\eta) = \|\boldsymbol{E}\boldsymbol{E}^\top\|^2 - \frac{1}{2}\|\boldsymbol{E}\|^4 \tag{4.26}$$

of ξ and η obtained by expressing f and f' in Eq. (4.23) in terms of ξ and η. Equation (4.24) is equivalent to $K(\xi,\eta) = 0$. Moreover, the following relation holds due to the singular nature of the essential matrix $\boldsymbol{E}$ [Kanatani et al. (2006)]:

$$K = \frac{\partial K}{\partial \xi} = \frac{\partial K}{\partial \eta} = 0. \tag{4.27}$$

This means that the function $K(\xi,\eta)$ takes its minimum at (ξ,η) *and* its minimum equals 0 (Fig. 4.4(a)). After some lengthy manipulations, we see that $K(\xi,\eta)$ is a fourth order polynomial in ξ and η, quadratic in each, in the form[2]

$$\begin{aligned} K(\xi,\eta) = {} & (\boldsymbol{k},\boldsymbol{F}\boldsymbol{k})^4\xi^2\eta^2 + 2(\boldsymbol{k},\boldsymbol{F}\boldsymbol{k})^2\|\boldsymbol{F}^\top\boldsymbol{k}\|^2\xi^2\eta + 2(\boldsymbol{k},\boldsymbol{F}\boldsymbol{k})^2\|\boldsymbol{F}\boldsymbol{k}\|^2\xi\eta^2 \\ & + \|\boldsymbol{F}^\top\boldsymbol{k}\|^4\xi^2 + \|\boldsymbol{F}\boldsymbol{k}\|^4\eta^2 + 4(\boldsymbol{k},\boldsymbol{F}\boldsymbol{k})(\boldsymbol{k},\boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k})\xi\eta + 2\|\boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{k}\|^2\xi \\ & + 2\|\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k}\|^2\eta + \|\boldsymbol{F}\boldsymbol{F}^\top\|^2 - \frac{1}{2}\Big((\boldsymbol{k},\boldsymbol{F}\boldsymbol{k})^2\xi\eta + \|\boldsymbol{F}^\top\boldsymbol{k}\|^2\xi \\ & + \|\boldsymbol{F}\boldsymbol{k}\|^2\eta + \|\boldsymbol{F}\|^2\Big)^2. \end{aligned} \tag{4.28}$$

Hence, the solution (ξ,η) can be obtained, in principle, by numerically searching for the minimum of this polynomial.

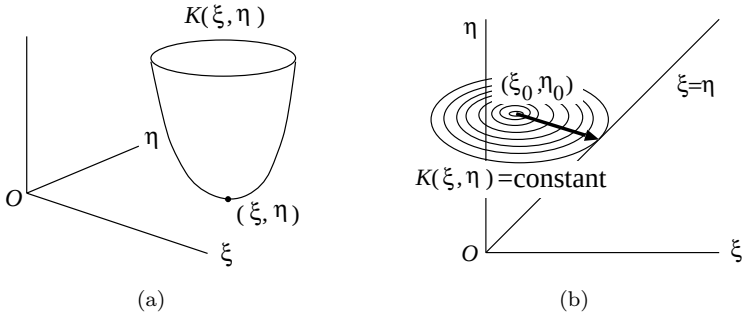
Here, we need to distinguish two types of algorithms. One is to see f and f' as independent variables and solve for them. Then, if the two cameras have the same focal lengths, the computed f and f' may not be equal, since $\boldsymbol{F}$ is computed from noisy point correspondences. Let us call this the *free focal length method*. The other is to force the equality $f = f'$ from the beginning and solve for them. In this case, Eq. (4.27) provide redundant constraints, since we have only one unknown. Let us call this the *fixed focal length method*.

4.4.2 *Free focal length method*

Instead of numerically searching for the minimum of Eq. (4.28), we can determine f and f' directly from Eq. (4.22). In fact, the problem of computing the focal lengths from $\boldsymbol{F}$ has attracted many researchers, and various formulas, all mathematically equivalent, have been presented in the past [Hartley (1992), Pan et al. (1995a,b), Newsam et al. (1996), Bougnoux (1998), Kanatani and Matsunaga (2000)]. Here, we describe the modified Bougnoux formula of Kanatani and Matsunaga (2000) (see Appendix D for the derivation). In the following, we define $\boldsymbol{k} = (0,\,0,\,1)^\top$.

Input: The fundamental matrix $\boldsymbol{F}$.
Output: The focal lengths f and f' of the two cameras.

[2]The EFNS procedure in Sec. 4.3.5 normalizes the vector $\boldsymbol{u}$ representing $\boldsymbol{F}$ to $\|\boldsymbol{u}\| = 1$. This means $\boldsymbol{F}$ is normalized to $\|\boldsymbol{F}\| = 1$.



(a) (b)

Fig. 4.4 (a) The focal lengths f and f' are determined by the point (ξ, η) at which the surface of $K(\xi, \eta)$ touches the $\xi\eta$-plane. (b) The point (ξ_0, η_0) is corrected to (ξ, η) on the line $\xi = \eta$ so that the increase of $K(\xi, \eta)$ is minimized.

(1) Let $\boldsymbol{e}$ and $\boldsymbol{e}'$ be the unit eigenvectors of $\boldsymbol{F}\boldsymbol{F}^\top$ and $\boldsymbol{F}^\top\boldsymbol{F}$, respectively, for the smallest eigenvalue.
(2) Compute the following ξ and η:

$$\xi = \frac{\|\boldsymbol{F}\boldsymbol{k}\|^2 - (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k})\|\boldsymbol{e}' \times \boldsymbol{k}\|^2/(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})}{\|\boldsymbol{e}' \times \boldsymbol{k}\|^2\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 - (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2},$$
$$\eta = \frac{\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 - (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k})\|\boldsymbol{e} \times \boldsymbol{k}\|^2/(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})}{\|\boldsymbol{e} \times \boldsymbol{k}\|^2\|\boldsymbol{F}\boldsymbol{k}\|^2 - (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2}. \tag{4.29}$$

(3) If $\xi \leq -1$ or $\eta \leq -1$, return the message *failure* and stop. Else return

$$f = \frac{f_0}{\sqrt{1+\xi}}, \qquad f' = \frac{f_0}{\sqrt{1+\eta}}. \tag{4.30}$$

Equations (4.30) are simply rewriting of Eqs. (4.25). The vectors $\boldsymbol{e}$ and $\boldsymbol{e}'$ computed in Step (1) represent the *epipoles* of $\boldsymbol{F}$. Namely, $(f_0 e_1/e_3, f_0 e_2/e_3)$ is the projection of the viewpoint of the second camera onto the first image, and $(f_0 e_1'/e_3', f_0 e_2'/e_3')$ is the projection of the viewpoint of the first camera onto the second image (see Eqs. (4.1)). If $e_3 = 0$ or $e_3' = 0$, the corresponding epipole is located at infinity in the image plane.

If the focal lengths f and f' are known to be equal, we may simply take the average $(f + f')/2$ as their values or replace ξ and η by their average $(\xi + \eta)/2$ in Step (2), but these are too ad hoc. A reasonable approach is to enforce the equality $\xi = \eta$ so that the increase of $K(\xi, \eta)$ is minimized (Fig. 4.4(b)). Let (ξ_0, η_0) be the solution given by Eqs. (4.30). The function $K(\xi, \eta)$ is Taylor expanded around (ξ_0, η_0) in the form

$$K(\xi, \eta) = \frac{1}{2}(\begin{pmatrix} \xi - \xi_0 \\ \eta - \eta_0 \end{pmatrix}, \boldsymbol{H} \begin{pmatrix} \xi - \xi_0 \\ \eta - \eta_0 \end{pmatrix}) + \cdots, \tag{4.31}$$

where $\cdots$ denotes terms of order 3 or higher in $\xi - \xi_0$ and $\eta - \eta_0$. From Eq. (4.28), the Hessian $\boldsymbol{H} = (H_{ij})$ has the form

$$\begin{aligned}
H_{11} &= 2(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^4\eta_0^2 + 4(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\|\boldsymbol{F}^\top\boldsymbol{k}\|^2\eta_0 + 2\|\boldsymbol{F}^\top\boldsymbol{k}\|^4 - \Big((\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\eta_0 + \|\boldsymbol{F}^\top\boldsymbol{k}\|^2\Big)^2, \\
H_{12} &= 4(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^4\xi_0\eta_0 + 4(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\Big(\|\boldsymbol{F}^\top\boldsymbol{k}\|^2\xi_0 + \|\boldsymbol{F}\boldsymbol{k}\|^2\eta_0\Big) + 4(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k}) \\
&\quad - \Big((\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\xi_0 + \|\boldsymbol{F}\boldsymbol{k}\|^2\Big)\Big((\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\eta_0 + \|\boldsymbol{F}^\top\boldsymbol{k}\|^2\Big) \\
&\quad - (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\Big((\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\xi_0\eta_0 + \|\boldsymbol{F}^\top\boldsymbol{k}\|^2\xi_0 + \|\boldsymbol{F}\boldsymbol{k}\|^2\eta_0 + \|\boldsymbol{F}\|^2\Big) = H_{21}, \\
H_{22} &= 2(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^4\xi_0^2 + 4(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\|\boldsymbol{F}\boldsymbol{k}\|^2\xi_0 + 2\|\boldsymbol{F}\boldsymbol{k}\|^4 - \Big((\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\xi_0 + \|\boldsymbol{F}\boldsymbol{k}\|^2\Big)^2.
\end{aligned} \tag{4.32}$$

If we ignore the high order terms $\cdots$, Eq. (4.31) is minimized subject to $\xi = \eta$ by

$$\xi = \eta = \frac{(H_{11} + H_{12})\xi_0 + (H_{22} + H_{12})\eta_0}{H_{11} + 2H_{12} + H_{22}}. \tag{4.33}$$

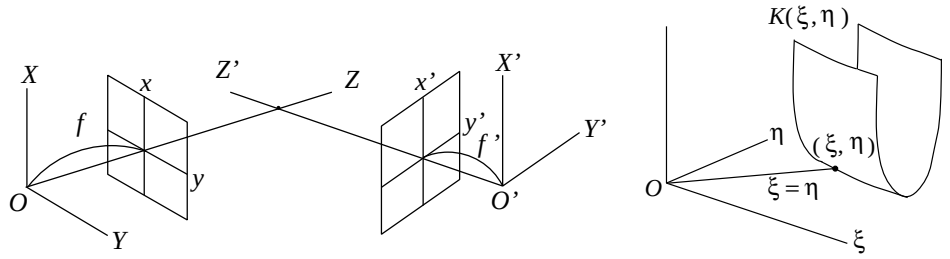
(Appendix E.) The corresponding focal lengths $f = f'$ are given by Eqs. (4.30).

4.4.3 *Fixed focal length method*

We can immediately see that the computation of Eqs. (4.29) fails when $K_{33} = (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k}) = 0$. If we recall Eqs. (4.1), we can view $\boldsymbol{k}$ as representing the image origin $(0, 0)$. From Eq. (4.2), we see that $(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k}) = 0$ implies that the image origin $(0, 0)$ of the first camera corresponds to the image origin $(0, 0)$ of the second, which means that the optical axes of the two camera intersect in the scene. This camera configuration is said to be *fixating* (Fig. 4.5(a)). The fact that the focal lengths cannot be determined in fixating camera configurations has been repeatedly pointed out by many researchers [Hartley (1992), Pan et al. (1995a,b), Newsam et al. (1996)]. Kanatani and Matsunaga (2000) showed that for fixating camera configurations the Hessian $\boldsymbol{H}$ in Eq. (4.31) degenerates to rank 1, so the surface defined by $K(\xi, \eta)$ becomes parabolic ("trough" shape) at (ξ, η) (Fig. 4.5(b)). Yet, the fixating camera configuration occurs very frequently in real situations.

This difficulty is resolved if the focal lengths f and f' are known to be equal [Brooks et al. (1998), Kanatani and Matsunaga (2000), Sturm (2001)]. Geometrically, the surface of $K(\xi, \eta)$ touches the $\xi\eta$-plane along the bottom of the trough. The equal focal lengths $\xi = \eta$ are obtained by the intersection of the bottom with the line $\xi = \eta$ (Fig. 4.5(b)). If ξ and η are equated, Eq. (4.28) reduces to the following fourth-order polynomial in ξ [Kanatani and Matsunaga (2000)]:

$$K(\xi) = a_1\xi^4 + a_2\xi^3 + a_3\xi^2 + a_4\xi + a_5, \tag{4.34}$$



(a) (b)

Fig. 4.5 (a) Fixating camera configuration. (b) The equal focal lengths $f = f'$ are determined by the intersection of the "bottom' of the surface of $K(\xi, \eta)$ and the line $\xi = \eta$.

$$
\begin{aligned}
a_1 &= \frac{1}{2}(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^4, \\
a_2 &= (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2(\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 + \|\boldsymbol{F}\boldsymbol{k}\|^2), \\
a_3 &= \frac{1}{2}(\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 - \|\boldsymbol{F}\boldsymbol{k}\|^2)^2 + (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})(4(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k}) - (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})\|\boldsymbol{F}\|^2), \\
a_4 &= 2(\|\boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{k}\|^2 + \|\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k}\|^2) - (\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 + \|\boldsymbol{F}\boldsymbol{k}\|^2)\|\boldsymbol{F}\|^2, \\
a_5 &= \|\boldsymbol{F}\boldsymbol{F}^\top\|^2 - \frac{1}{2}\|\boldsymbol{F}\|^4.
\end{aligned} \tag{4.35}
$$

The minimum of $K(\xi)$ is obtained by solving the following cubic equation in ξ:

$$K'(\xi) = 4a_1\xi^3 + 3a_2\xi^2 + 2a_3\xi + a_4 = 0. \tag{4.36}$$

The corresponding $f = f'$ are given by Eqs. (4.30). This solution is valid as long as $f = f'$ irrespective of whether the camera configuration is fixating or not. In a strict fixating configuration, we have $(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k}) = 0$, so $a_1 = a_2 = 0$ and hence Eq. (4.36) reduces to $2a_3\xi + a_4 = 0$ with the solution

$$\xi = -\frac{2(\|\boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{k}\|^2 + \|\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k}\|^2) - (\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 + \|\boldsymbol{F}\boldsymbol{k}\|^2)\|\boldsymbol{F}\|^2}{(\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 - \|\boldsymbol{F}\boldsymbol{k}\|^2)^2}. \tag{4.37}$$

Thus, we obtain the following algorithm:

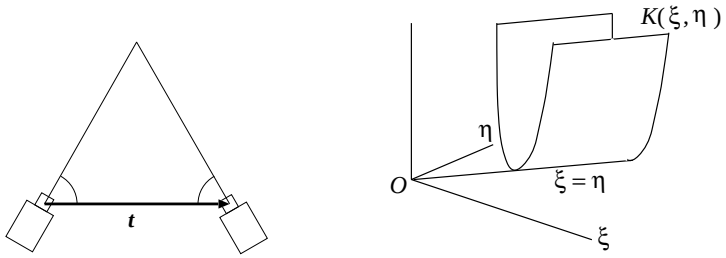
Input: The fundamental matrix $\boldsymbol{F}$.
Output: The focal lengths $f = f'$ of the two cameras.

(1) If $(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k}) \approx 0$, compute the ξ in Eq. (4.37), and go to Step (3).
(2) Else, solve the cubic equation $K'(\xi) = 0$ in Eq. (4.36). If it has a single real zero, let it be ξ. If it has three real zeros $\xi_1 \le \xi_2 \le \xi_3$, let

$$\xi = \begin{cases} \xi_3 & \xi_2 \le -1 \ \text{ or } \ K(\xi_1) < 0 \ \text{ or } \ K(\xi_3) \le K(\xi_1) \\ \xi_1 & 0 \le K(\xi_1) < K(\xi_3) \\ -1 & \text{otherwise} \end{cases}. \tag{4.38}$$

(3) If $\xi \le -1$, return the message *failure* and stop. Else return

$$f = f' = \frac{f_0}{\sqrt{1+\xi}}. \tag{4.39}$$



(a) (b)

Fig. 4.6 (a) Symmetric camera configuration. (b) The surface of $K(\xi, \eta)$ touches the $\xi\eta$-plane along $\xi = \eta$.

4.4.4 *Issues*

If the camera configuration is strictly fixating, we have no choice but to use Eq. (4.37). In a practical situation, however, the camera configuration may be nearly fixating but not exactly fixating. In a non-fixating situation, we have two solutions; one obtained using the free focal length method followed by forcing the equality $f = f'$ by Eq. (4.33); the other using the fixed focal length method. Having two solutions decreases the chance of computational failure. In fact, the inside of the square roots in Eqs. (4.30) and Eq. (4.39) frequently becomes negative. This may be due to the inaccurate correspondence point detection but also because the camera model is not accurate enough, in particular, the principal point is not at the center of the image [Hartley and Silpa-Anan (2002)].

There is another source of computational failure, however. It has been pointed out that the focal lengths cannot be computed even for $f = f'$ if the camera configuration is fixating and *symmetric* or the two optical axes and the baseline make an isosceles triangle (including parallel optical axes) [Brooks et al. (1998), Kanatani and Matsunaga (2000), Sturm (2001)] (Fig. 4.6(a)). In such configurations, the function $K(\xi, \eta)$ in Eq. (4.28) is symmetric with respect to ξ and η, i.e., $K(\xi, \eta) = K(\eta, \xi)$. Hence, the surface of $K(\xi, \eta)$ touches the $\xi\eta$-plane along the line $\xi = \eta$ [Kanatani and Matsunaga (2000)] (Fig. 4.6(b)). Unfortunately, this is the most frequently encountered situation in real circumstances; people unconsciously place the cameras in such a configurations.

Thus, in real situations, if one of the two focal length computation methods returns a real solution while the other fails, we can use the real one. If both fail, there is no other way but to use some empirical value for f and f'. If both returns real values, however, which solution should be used? Kanatani et al. (2006) reported that even though images are taken without changing the focus or the zoom, the free focal length method is often more accurate if the camera configuration is non-fixating to some degree. However, setting an appropriate threshold for it is very difficult. A practical solution may be to retain both solutions and compute 3-D reconstruction in two ways. In the end, we choose a better one according to

some criterion, e.g., we may check if corners known to be right angles are correctly reconstructed. If no such criterion exists, we may choose the reconstruction for which the reprojection error is smaller (the computation is described in Sec. 4.5.2).

4.5 3-D Reconstruction Procedure

4.5.1 *Computation of motion parameters*

Once the focal lengths $\{f,\ f'\}$ are computed, it is easy to compute the motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$ that satisfy Eq. (4.22). The following procedure is well known [Kanatani (1993)]:

Input: The fundamental matrix $\boldsymbol{F}$, the focal lengths f and f', and the image coordinates (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$, of corresponding points.
Output: The motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$.

(1) Compute from $\boldsymbol{F}$ the essential matrix $\boldsymbol{E}$ in Eq. (4.23).
(2) Let $\boldsymbol{t}$ be the unit eigenvector of $\boldsymbol{E}\boldsymbol{E}^\top$ for the smallest eigenvalue.
(3) Change the sign of $\boldsymbol{t}$ if

$$\sum_{\alpha=1}^{N} |\boldsymbol{t}, \boldsymbol{x}_\alpha, \boldsymbol{E}\boldsymbol{x}'_\alpha| < 0, \tag{4.40}$$

where $|\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}|$ denotes the scalar triple product of vectors $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{c}$.
(4) Compute the singular value decomposition (SVD) of $-\boldsymbol{t} \times \boldsymbol{E}$ in the form

$$-\boldsymbol{t} \times \boldsymbol{E} = \boldsymbol{U}\mathrm{diag}(\sigma_1, \sigma_2, \sigma_3)\boldsymbol{V}^\top. \tag{4.41}$$

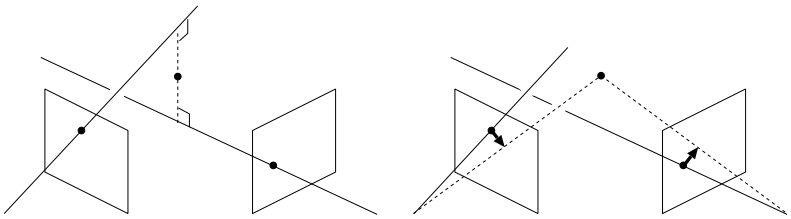
(5) Compute the rotation $\boldsymbol{R}$ as follows:

$$\boldsymbol{R} = \boldsymbol{U}\mathrm{diag}(1, 1, \det(\boldsymbol{U}\boldsymbol{V}^\top))\boldsymbol{V}^\top. \tag{4.42}$$

If f and f' are exactly computed so that Eq. (4.22) is satisfied, we have $\boldsymbol{E} = \boldsymbol{t}\times\boldsymbol{R}$. Hence, $\boldsymbol{E}\boldsymbol{t} = \boldsymbol{0}$, i.e., $\boldsymbol{t}$ is the eigenvector of $\boldsymbol{E}$ for eigenvalue 0. Step (2) ensures a stable computation even when f and f' are not exact. However, eigenvectors have sign indeterminacy. Step (3) ensures that on the plane passing through $\boldsymbol{x}_\alpha$, $\boldsymbol{t}$, and $\boldsymbol{R}\boldsymbol{x}'_\alpha$, the vectors $\boldsymbol{x}_\alpha$ and $\boldsymbol{R}\boldsymbol{x}'_\alpha$ are on the same side of $\boldsymbol{t}$. This is easily seen from the identity $|\boldsymbol{t}, \boldsymbol{x}_\alpha, \boldsymbol{E}\boldsymbol{x}'_\alpha| = (\boldsymbol{t} \times \boldsymbol{x}_\alpha, \boldsymbol{t} \times \boldsymbol{R}\boldsymbol{x}'_\alpha)$. Step (4) returns the rotation $\boldsymbol{R}$ that minimizes $\|\boldsymbol{t} \times \boldsymbol{R} - \boldsymbol{E}\|$ [Kanatani (1993)].

4.5.2 *Optimal triangulation*

Having obtained the focal lengths $\{f,\ f'\}$ and the motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$, we can compute the 3-D location of each point. In fact, if we know the intrinsic and extrinsic camera parameters, we can define for each point its line of sight. The 3-D location $(X_\alpha, Y_\alpha, Z_\alpha)$ is the intersection of the lines of sight of (x_α, y_α) and (x'_α, y'_α).



(a) (b)

Fig. 4.7 Triangulation. (a) The mid-point method. (b) Optimal correction.

Due to noise in observation, however, the lines of sight need not meet in the scene. A naive idea is to regard the "midpoint" of the shortest line segment connecting the two lines of sight as the intersection (Fig. 4.7(a)). The best way is to displace the corresponding points so that their lines of sight intersect in such a way that the amount of the displacement is minimum (Fig. 4.7(b)). Mathematically, this amounts to correct (x_α, y_α) and (x'_α, y'_α) to $(\hat{x}_\alpha, \hat{y}_\alpha)$ and $(\hat{x}'_\alpha, \hat{y}'_\alpha)$ in such a way that the reprojection error

$$E_\alpha = (x_\alpha - \hat{x}_\alpha)^2 + (y_\alpha - \hat{y}_\alpha)^2 + (x'_\alpha - \hat{x}'_\alpha)^2 + (y'_\alpha - \hat{y}'_\alpha)^2 \tag{4.43}$$

is minimized. As noted in Sec. 4.3.1, this is maximum likelihood (ML) estimation if the noise in (x_α, y_α) and (x'_α, y'_α) is independent and identical Gaussian variables of mean 0. Minimization of Eq. (4.43) is exactly the same as Sec. 4.3.4 except that the fundamental matrix is now regarded as given [Kanatani et al. (2008)]:

Input: The focal lengths $\{f,\ f'\}$, the motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$, and the image coordinates (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$, of corresponding points.

Output: Corrected image coordinates $(\hat{x}_\alpha, \hat{y}_\alpha)$, $(\hat{x}'_\alpha, \hat{y}'_\alpha)$, $\alpha = 1, ..., N$, and the reprojection error E.

(1) Recompute the essential matrix $\boldsymbol{E}$ as follows:

$$\boldsymbol{E} = \boldsymbol{t} \times \boldsymbol{R}. \tag{4.44}$$

(2) Compute the following 9-D vector $\boldsymbol{u}$:

$$\boldsymbol{u} = (E_{11},\ E_{12},\ E_{13},\ E_{21},\ E_{22},\ E_{23},\ E_{31},\ E_{32},\ E_{33})^\top. \tag{4.45}$$

(3) Do the following computation for $\alpha = 1, ..., N$:

(a) Let $E^0_\alpha = \infty$ (a sufficiently large number), and initialize $\hat{x}_\alpha$, $\hat{y}_\alpha$, $\hat{x}'_\alpha$, $\hat{y}'_\alpha$, $\tilde{x}_\alpha$, $\tilde{y}_\alpha$, $\tilde{x}'_\alpha$, and $\tilde{y}'_\alpha$ as follows:

$$\hat{x}_\alpha = x_\alpha, \quad \hat{y}_\alpha = y_\alpha, \quad \hat{x}'_\alpha = x'_\alpha, \quad \hat{y}'_\alpha = y'_\alpha,$$

$$\tilde{x}_\alpha = \tilde{y}_\alpha = \tilde{x}'_\alpha = \tilde{y}'_\alpha = 0. \tag{4.46}$$

(b) Compute the following 9-D vectors $\boldsymbol{\xi}_\alpha$ and 9×9 matrices $V_0[\boldsymbol{\xi}_\alpha]$:

$$\boldsymbol{\xi}_\alpha = \begin{pmatrix} \hat{x}_\alpha \hat{x}'_\alpha + \hat{x}'_\alpha \tilde{x}_\alpha + \hat{x}_\alpha \tilde{x}'_\alpha \\ \hat{x}_\alpha \hat{y}'_\alpha + \hat{y}'_\alpha \tilde{x}_\alpha + \hat{x}_\alpha \tilde{y}'_\alpha \\ f'(\hat{x}_\alpha + \tilde{x}_\alpha) \\ \hat{y}_\alpha \hat{x}'_\alpha + \hat{x}'_\alpha \tilde{y}_\alpha + \hat{y}_\alpha \tilde{x}'_\alpha \\ \hat{y}_\alpha \hat{y}'_\alpha + \hat{y}'_\alpha \tilde{y}_\alpha + \hat{y}_\alpha \tilde{y}'_\alpha \\ f'(\hat{y}_\alpha + \tilde{y}_\alpha) \\ f(\hat{x}'_\alpha + \tilde{x}'_\alpha) \\ f(\hat{y}'_\alpha + \tilde{y}'_\alpha) \\ ff' \end{pmatrix}, \tag{4.47}$$

$$V_0[\boldsymbol{\xi}_\alpha] = \begin{pmatrix} \hat{x}_\alpha^2 + \hat{x}_\alpha'^2 & \hat{x}'_\alpha \hat{y}'_\alpha & f\hat{x}'_\alpha & \hat{x}_\alpha \hat{y}_\alpha & 0 & 0 & f'\hat{x}_\alpha & 0 & 0 \\ \hat{x}'_\alpha \hat{y}'_\alpha & \hat{x}_\alpha^2 + \hat{y}_\alpha'^2 & f\hat{y}'_\alpha & 0 & \hat{x}_\alpha \hat{y}_\alpha & 0 & 0 & f'\hat{x}_\alpha & 0 \\ f\hat{x}'_\alpha & f\hat{y}'_\alpha & ff' & 0 & 0 & 0 & 0 & 0 & 0 \\ \hat{x}_\alpha \hat{y}_\alpha & 0 & 0 & \hat{y}_\alpha^2 + \hat{x}_\alpha'^2 & \hat{x}'_\alpha \hat{y}'_\alpha & f\hat{x}'_\alpha & f'\hat{y}_\alpha & 0 & 0 \\ 0 & \hat{x}_\alpha \hat{y}_\alpha & 0 & \hat{x}'_\alpha \hat{y}'_\alpha & \hat{y}_\alpha^2 + \hat{y}_\alpha'^2 & f\hat{y}'_\alpha & 0 & f'\hat{y}_\alpha & 0 \\ 0 & 0 & 0 & f\hat{x}'_\alpha & f\hat{y}'_\alpha & ff' & 0 & 0 & 0 \\ f'\hat{x}_\alpha & 0 & 0 & f'\hat{y}_\alpha & 0 & 0 & ff' & 0 & 0 \\ 0 & f'\hat{x}_\alpha & 0 & 0 & f'\hat{y}_\alpha & 0 & 0 & ff' & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.48}$$

(c) Update $\tilde{x}_\alpha$, $\tilde{y}_\alpha$, $\tilde{x}'_\alpha$, and $\tilde{y}'_\alpha$ as follows:

$$\begin{pmatrix} \tilde{x}_\alpha \\ \tilde{y}_\alpha \end{pmatrix} \leftarrow \frac{(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} \begin{pmatrix} u_1 & u_2 & u_3 \\ u_4 & u_5 & u_6 \end{pmatrix} \begin{pmatrix} \hat{x}'_\alpha \\ \hat{y}'_\alpha \\ f' \end{pmatrix},$$

$$\begin{pmatrix} \tilde{x}'_\alpha \\ \tilde{y}'_\alpha \end{pmatrix} \leftarrow \frac{(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} \begin{pmatrix} u_1 & u_4 & u_7 \\ u_2 & u_5 & u_8 \end{pmatrix} \begin{pmatrix} \hat{x}_\alpha \\ \hat{y}_\alpha \\ f \end{pmatrix}. \tag{4.49}$$

(d) Update $\hat{x}_\alpha$, $\hat{y}_\alpha$, $\hat{x}'_\alpha$, and $\hat{y}'_\alpha$ as follows:

$$\hat{x}_\alpha \leftarrow x_\alpha - \tilde{x}_\alpha, \quad \hat{y}_\alpha \leftarrow y_\alpha - \tilde{y}_\alpha,$$

$$\hat{x}'_\alpha \leftarrow x'_\alpha - \tilde{x}'_\alpha, \quad \hat{y}'_\alpha \leftarrow y'_\alpha - \tilde{y}'_\alpha. \tag{4.50}$$

(e) Compute the reprojection error E_α as follows:

$$E_\alpha = \tilde{x}_\alpha^2 + \tilde{y}_\alpha^2 + \tilde{x}'^{\,2}_\alpha + \tilde{y}'^{\,2}_\alpha. \tag{4.51}$$

(f) If $E_\alpha \not\approx E_\alpha^0$, let $E_\alpha^0 \leftarrow E_\alpha$ and go back to Step (3b).

(4) Return $(\hat{x}_\alpha, \hat{y}_\alpha)$, $(\hat{x}'_\alpha, \hat{y}'_\alpha)$, $\alpha = 1, ..., N$, and the following (total) reprojection error E:

$$E = \sum_{\alpha=1}^{N} E_\alpha. \tag{4.52}$$

If we use the focal lengths $\{f,\ f'\}$ computed by the free focal length method (without failure), the resulting values of $(\hat{x}_\alpha, \hat{y}_\alpha)$ and $(\hat{x}'_\alpha, \hat{y}'_\alpha)$ are identical to those computed in Sec. 4.3.4, because the fundamental matrix $\boldsymbol{F}$ was computed to minimize the reprojection error E in Eq. (4.52). If the equality $f = f'$ is enforced, however, the results are not necessarily the same, and we need the recomputation as described above. If we have retained two solutions, one by the free focal length method followed by enforcement of $f = f'$, the other by the fixed focal length method, the selection can be done at this stage: we choose the one that has a smaller reprojection error E computed here.

4.5.3 *3-D location computation*

A 3-D point (X, Y, Z) is projected to a point (x, y) on the image plane by a general perspective camera in the form

$$\begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix} \simeq \boldsymbol{P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \tag{4.53}$$

where the symbol $\simeq$ means that both sides are equal up to a nonzero constant. Here, $\boldsymbol{P}$ is a 3×4 matrix called the *projection matrix* or the *camera matrix*. If the camera coordinate system is placed at $\boldsymbol{t}$ and rotated by $\boldsymbol{R}$ relative to a fixed world coordinate system, $\boldsymbol{P}$ has form

$$\boldsymbol{P} \simeq \boldsymbol{K} \left(\boldsymbol{R}^\top \ -\boldsymbol{R}^\top \boldsymbol{t} \right), \tag{4.54}$$

where $\boldsymbol{K}$ is an upper triangular matrix, called the *matrix of intrinsic parameters*, in the following form [Hartley and Zisserman (2000)]:

$$\boldsymbol{K} = \begin{pmatrix} \gamma f/f_0 & \gamma \tan\theta & u_0/f_0 \\ & f/f_0 & v_0/f_0 \\ & & 1 \end{pmatrix}. \tag{4.55}$$

Here, f is the focal length, γ is the aspect ratio, θ is the skew angle (defined to be 0 in the skewless case), and (u_0, v_0) is the principal point location. As discussed in Sec. 4.4.1, we need to assume that $\gamma = 1$, $\theta = 0$, and $(u_0, v_0) = (0, 0)$ for doing 3-D reconstruction from two views. Thus, if the two cameras have focal length f and f' with motion parameters $\{\boldsymbol{t},\ \boldsymbol{R}\}$ and if the first camera is identified with the world coordinate system, the projection matrices of the first and the second cameras can be written up scale as follows:

$$\boldsymbol{P} \simeq \mathrm{diag}(1, 1, \frac{f_0}{f}) \left(\boldsymbol{I}\ \boldsymbol{0} \right), \qquad \boldsymbol{P'} \simeq \mathrm{diag}(1, 1, \frac{f_0}{f'}) \left(\boldsymbol{R}^\top \ -\boldsymbol{R}^\top \boldsymbol{t} \right). \tag{4.56}$$

From Eq. (4.53), we obtain the following projection relations:

$$\begin{cases} x = f_0 \dfrac{P_{11}X + P_{12}Y + P_{13}Z + P_{14}}{P_{31}X + P_{32}Y + P_{33}Z + P_{34}}, \\ y = f_0 \dfrac{P_{21}X + P_{22}Y + P_{23}Z + P_{24}}{P_{31}X + P_{32}Y + P_{33}Z + P_{34}}, \end{cases} \quad \begin{cases} x' = f_0 \dfrac{P'_{11}X + P'_{12}Y + P'_{13}Z + P'_{14}}{P'_{31}X + P'_{32}Y + P'_{33}Z + P'_{34}}, \\ y' = f_0 \dfrac{P'_{21}X + P'_{22}Y + P'_{23}Z + P'_{24}}{P'_{31}X + P'_{32}Y + P'_{33}Z + P'_{34}}. \end{cases} \tag{4.57}$$

From these, we obtain the following linear equations:

$$\begin{pmatrix} xP_{31} - f_0P_{11} & xP_{32} - f_0P_{12} & xP_{33} - f_0P_{13} \\ yP_{31} - f_0P_{21} & yP_{32} - f_0P_{22} & yP_{33} - f_0P_{23} \\ x'P'_{31} - f_0P'_{11} & x'P'_{32} - f_0P'_{12} & x'P'_{33} - f_0P'_{13} \\ y'P'_{31} - f_0P'_{21} & y'P'_{32} - f_0P'_{22} & y'P'_{33} - f_0P'_{23} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = - \begin{pmatrix} xP_{34} - f_0P_{14} \\ yP_{34} - f_0P_{24} \\ x'P'_{34} - f_0P'_{14} \\ y'P'_{34} - f_0P'_{24} \end{pmatrix}. \tag{4.58}$$

These provide four equations for three unknowns X, Y, and Z. We can choose from among them any three and solve them. Alternatively, we can solve the four redundant equations by least squares (Appendix F). Both give the same solution if we use $(\hat{x}_\alpha, \hat{y}_\alpha)$ and $(\hat{x}'_\alpha, \hat{y}'_\alpha)$ as image coordinates, because the procedure in Sec. 4.5.2 is done for that purpose in an optimal manner.

However, we need to check the *cheirality* [Hartley and Zisserman (2000)]. The perspective projection described by Eq. (4.53) stipulates that a 3-D point is projected to the intersection of the ray it defines with the image plane, but no constraint is imposed on the 3-D point to be *in front* of the camera. The condition in Eq. (4.40) only ensures that the 3-D points are both in front of the two cameras (the correct shape) or both behind them (the mirror image shape). This ambiguity originates from the sign indeterminacy of the fundamental matrix $\boldsymbol{F}$ in Eq. (4.2); Eq. (4.40) merely choose the sign of $\boldsymbol{t}$ compatible to $\boldsymbol{F}$. Thus, after computing the 3-D locations $(X_\alpha, Y_\alpha, Z_\alpha)$ of all the points, we reverse the sign of $\boldsymbol{t}$ and all $(X_\alpha, Y_\alpha, Z_\alpha)$ if

$$\sum_{\alpha=1}^{N} \mathrm{sgn}(Z_\alpha) < 0, \tag{4.59}$$

where $\mathrm{sgn}(x)$ is the signature function that returns 1, -1, and 0 for $x > 0$, $x < 0$, and $x = 0$, respectively. We introduce $\mathrm{sgn}(x)$ because if we require $\sum_{\alpha=1}^{N} Z_\alpha > 0$, the judgment may be reversed when a very large depth $Z_\alpha \approx \infty$ with nearly parallel lines of sight may be computed to be $Z_\alpha \approx -\infty$.

The reconstructed 3-D shape has scale ambiguity, as we discussed in Sec. 4.4.1, so we introduced the scale normalization $\|\boldsymbol{t}\| = 1$. This ambiguity is resolved if some metric information is available. For example, if the camera translation $\|\boldsymbol{t}\|$ is known, the computed 3-D shape is magnified by $\|\boldsymbol{t}\|$. If the distance between two 3-D points, say (X_1, Y_1, Z_1) and (X_2, Y_2, Z_2), is known to be d_{12}, we can recover the correct scale by multiplying all $(X_\alpha, Y_\alpha, Z_\alpha)$ and $\boldsymbol{t}$ by $d_{12}/\sqrt{(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2}$. Instead of taking the first camera as a reference, we can also view the first camera as displaced by $\boldsymbol{t}_c$ and rotated

by $\boldsymbol{R}_c$ relative to a fixed world coordinate system. The 3-D locations $(\bar{X}_\alpha, \bar{Y}_\alpha, \bar{Z}_\alpha)$ with respect to this coordinate system are given by

$$\begin{pmatrix} \bar{X}_\alpha \\ \bar{Y}_\alpha \\ \bar{Z}_\alpha \end{pmatrix} = \boldsymbol{R}_c \begin{pmatrix} X_\alpha \\ Y_\alpha \\ Z_\alpha \end{pmatrix} + \boldsymbol{t}_c \quad \Big(= \begin{pmatrix} \boldsymbol{R}_c & \boldsymbol{t}_c \end{pmatrix} \begin{pmatrix} X_\alpha \\ Y_\alpha \\ Z_\alpha \\ 1 \end{pmatrix} \Big). \tag{4.60}$$

4.6 Concluding Remarks

3-D reconstruction from two views has intensively been studied since early 1980s as one of the major problems of computer vision. After nearly 30 years of its history, however, efforts are still being made to improve the computational accuracy and efficiency. This chapter has presented the state-of-the-art technology of this problem.

Since we only need to take two pictures, two view reconstruction is one of the most useful tools in computer vision applications. However, we must also realize that limiting to two views imposes severe restrictions on its applicability. Computer vision practitioners are frequently confounded by the failure of 3-D reconstruction due to the lack of understanding such restrictions. Here, we describe typical *failure modes*:

Infinitely far away scene: If the scene is infinitely far away, the two views are essentially the same (up to camera rotation), so no 3-D information is obtained. In practice, this is the case if the scene is sufficiently far away as compared to the translation $\boldsymbol{t}$,

Planar scene: The fundamental matrix $\boldsymbol{F}$ cannot be computed uniquely if all the points are coplanar in 3-D (Fig. 4.8(a)). This is a typical instance of the critical surface mentioned in Sec. 4.3.1; that plane and a plane passing through the two viewpoints O and O' constitute a degenerate quadric surface. However, it is possible to compute 3-D structure and motion by exploiting the geometry of projective transformation, or "homography", instead of the epipolar geometry. [Kanatani (1993, 1996), Hartley and Zisserman (2000)].

Box scene: The fundamental matrix $\boldsymbol{F}$ cannot be computed uniquely if we observe the eight vertices of a cube (Fig. 4.8(b)). This is also an instance of the critical surface; there always exists a quadric surface that passes through the eight vertices and arbitrarily placed viewpoints O and O'. This configuration usually does not occur in real circumstances, yet this frequently happens in simulation, because people tend to test their algorithms using this configuration. In simulation, we should use general configurations, avoiding too regular or exceptional settings.

Pure rotation: No 3-D information is obtained if we observe the scene by rotating the camera around the lens center (Fig. 4.8(c)). This is because the lines of

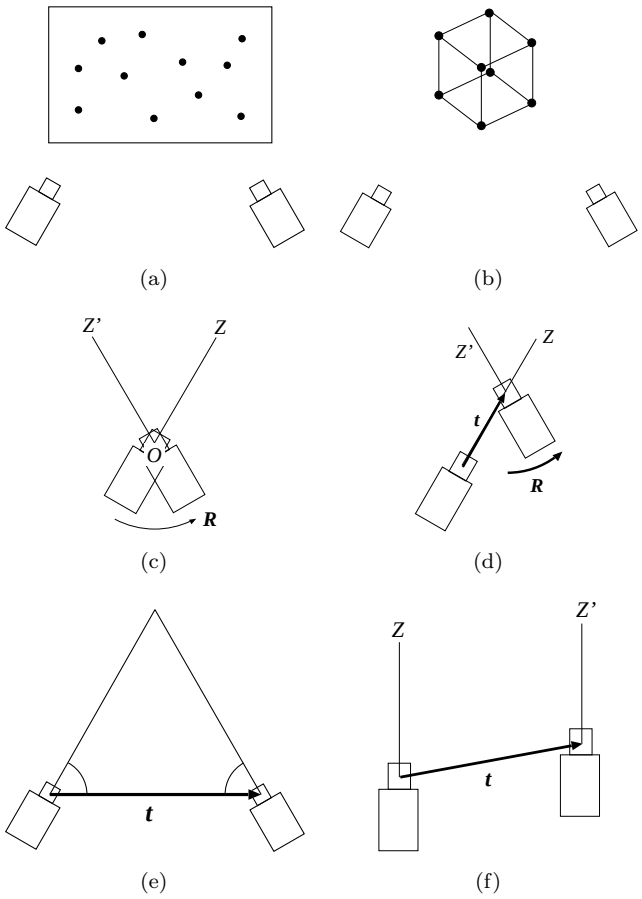


Fig. 4.8 Typical failure modes of two view 3-D reconstruction. (a) Planar scene. (b) Box scene. (c) Pure rotation. (d) Advancing camera. (e) Symmetric cameras. (f) Pure translation.

sight coming through the lens center are the same before and after the camera rotation. Hence, the fundamental matrix cannot be computed. However, the camera rotation can be determined by computing the homography [Hartley and Zisserman (2000)].

Advancing camera: If one camera is moved along its optical axis and then rotated (Fig. 4.8(d)), the focal lengths f and f' cannot be computed by the free-focal length method. This is because the two cameras are in a fixating configuration with the fixating point *at the lens center of the second camera*. This fact is easily overlooked in a real situation. In such a case, the focal lengths $f = f'$ can be computed if the fixed focal length method is used as long as the rotation of the second camera is nonzero. However, a care is necessary: the 3-D location of a point at the center of the first image cannot be determined, because it is the epipole of the first camera, i.e., the projection of the viewpoint of the second

camera. In fact, a 3-D point P on the line connecting the viewpoints O and O' of the two cameras cannot be "triangulated" if O, O', and P are collinear. In actual computation, the rank of the matrix in Eq. (4.58) drops to 2, and the solution is indeterminate. Points near the center of the first image can be triangulated, but their computed 3-D locations suffer considerable degradation in accuracy.

Symmetric cameras: As mentioned in Sec. 4.4.4, the focal lengths f and f' cannot be computed if the camera configuration is symmetric (Fig. 4.8(e)). We cannot compute f and f' even if we impose the equality $f = f'$. Of course, 3-D reconstruction is possible if the focal lengths f and f' are known. In fact, this is a standard camera configuration of stereo vision.

Pure translation: The focal lengths f and f' cannot be computed if one camera is translated with the optical axis direction fixed (Fig. 4.8(f)). This is a fixating configuration with the fixation point at infinity. We cannot compute f and f' even if we impose the equality $f = f'$, because the translation $\boldsymbol{t}$ and the two optical axes make an (infinitely elongated) isosceles triangle with the vertex at infinity. Again, 3-D reconstruction is possible if the focal lengths f and f' are known; this is also a standard camera configuration of stereo vision.

Besides avoiding these failure modes, caution is necessary when black box tools are used. For example, the numerical values of the fundamental matrix $\boldsymbol{F}$ depends on the image coordinate system used as well as the scaling constant f_0. For example, if we regard the upper-left corner of the image as the origin $(0, 0)$, take the x-axis direction rightward and the y-axis direction upward, and let $f_0 = 1$, which is a standard convention in image processing applications, the fundamental matrix $\boldsymbol{F}$ has a completely different numerical value. Some authors, e.g., Hartley and Zisserman (2000), define the fundamental matrix $\boldsymbol{F}$ to be the transpose of the one defined in this chapter, so that the epipolar equation is written, instead of Eq. (4.2), as $(\bar{\boldsymbol{x}}'_\alpha, \boldsymbol{F}\bar{\boldsymbol{x}}_\alpha) = 0$. Thus, users must correctly understand the definitions and conventions used in ready-made tools. Then, the theories and computational techniques described in this chapter will allow users to do two view reconstruction in many practical applications in useful ways.

Finally, it should be stressed that the quality of 3-D reconstruction critically depends on the quality of correspondence detection. The standard procedure is to first detect typical points called *feature points*, *corner points*, *points of interest*, or *keypoints* by image filters such as the Harris operator [Harris and Stephens (1988)], SUSAN [Smith and Brady (1997)], MSER [Matas, et. al (2002)] and SIFT [Lowe (2004)] in two images independently and then match them using appropriate measures such as intensity correlations, patch similarities, and feature descriptors. Finally, false matches, or *outliers*, that are incompatible with the epipolar geometry are removed by a voting procedure such as RANSAC [Fischer and Bolles (1981)] and LMedS [Rousseeuw and Leroy (1978)], e.g. see Zhang et al. (1995) and Kanazawa

and Kanatani (2004). Today, the use of video images is common, and feature points trackers such as the Kanade-Lucas-Tomasi algorithm (KLT) [Tomasi and Kanade (1991)] is widely used. Then, two frames are selected from the sequence, and those points continually tracked between them are used for 3-D reconstruction. However, correct correspondence detection is still a difficult task, and further progress is required for reliable 3-D reconstruction.

Appendix

A. Least squares

The simplest scheme for fundamental matrix computation is the following least squares, also known as *algebraic distance minimization* and *Hartley's 8-point algorithm* [Hartley (1997)].

Input: The image coordinates (x_α, y_α), (x'_α, y'_α), $\alpha = 1, ..., N$ (≥ 8), of corresponding points.
Output: The fundamental matrix $\boldsymbol{F}$.

(1) Represent the corresponding points (x_α, y_α) and (x'_α, y'_α), $\alpha = 1, ..., N$, by the following 9-D vectors:

$$\boldsymbol{\xi}_\alpha = (x_\alpha x'_\alpha,\ x_\alpha y'_\alpha,\ f_0 x_\alpha,\ y_\alpha x'_\alpha,\ y_\alpha y'_\alpha,\ f_0 y_\alpha,\ f_0 x'_\alpha,\ f_0 y'_\alpha,\ f_0^2)^\top. \tag{4.61}$$

(2) Let $\boldsymbol{u}$ be the 9-D unit eigenvector of the 9×9 matrix

$$\boldsymbol{M} = \sum_{\alpha=1}^{N} \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top, \tag{4.62}$$

for the smallest eigenvalue.
(3) Return the following fundamental matrix:

$$\boldsymbol{F} = \begin{pmatrix} u_1 & u_2 & u_3 \\ u_4 & u_5 & u_6 \\ u_7 & u_8 & u_9 \end{pmatrix}. \tag{4.63}$$

If $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ are converted in to the 9-D vector in Eq. (4.61) and if the fundamental matrix $\boldsymbol{F}$ is represented by the 9-D vector $\boldsymbol{u}$ in Eq. (4.4), the epipolar equation $(\boldsymbol{x}_\alpha, \boldsymbol{F}\boldsymbol{x}'_\alpha) = 0$ is equivalently written as

$$(\boldsymbol{u}, \boldsymbol{\xi}_\alpha) = 0. \tag{4.64}$$

Mathematically, the vector $\boldsymbol{\xi}_\alpha$ in Eq. (4.61) is called the *Kronecker product* of $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ (multiplied by f_0^2). For noisy data, Eq. (4.64) is not exactly satisfied, so we seek the vector $\boldsymbol{u}$ that minimizes

$$J_{\mathrm{LS}} = \sum_{\alpha=1}^{N} (\boldsymbol{u}, \boldsymbol{\xi}_\alpha)^2 = \sum_{\alpha=1}^{N} \boldsymbol{u}^\top \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top \boldsymbol{u} = (\boldsymbol{u}, \boldsymbol{M}\boldsymbol{u}), \tag{4.65}$$

where $\boldsymbol{M}$ is defined by Eq. (4.62). If we normalized $\boldsymbol{u}$ to a unit vector ($\|\boldsymbol{u}\| = 1$) to remove the scale indeterminacy of $\boldsymbol{F}$, the quadratic form in Eq. (4.65) is minimized by the unit eigenvector of $\boldsymbol{M}$ for the smallest eigenvalue.

The matrix $\boldsymbol{F}$ thus computed does not necessarily satisfy the rank constraint $\det \boldsymbol{F} = 0$. A simple method for remedy is to compute the singular value decomposition (SVD) of $\boldsymbol{F}$ and replace the smallest singular value by 0 [Hartley (1997)]. More sophisticated methods exist for optimally correcting $\boldsymbol{F}$ so that $\det \boldsymbol{F} = 0$ holds based on statistical analysis of noise [Kanatani and Sugaya (2007a)]. However, such sophistication is not necessary in practice, because the least squares solution itself has only limited accuracy [Kanatani and Sugaya (2007a,c)].

B. Taubin method

If the noise in x_α, y_α, x'_α, and y'_α is regarded as independent Gaussian variables of mean 0 and standard deviation σ, the vector $\boldsymbol{\xi}_\alpha$ in Eq. (4.61) has the following covariance matrix [Kanatani and Sugaya (2007a,c)]:

$$V[\boldsymbol{\xi}_\alpha] = \sigma^2 V_0[\boldsymbol{\xi}_\alpha], \tag{4.66}$$

$$V_0[\boldsymbol{\xi}_\alpha] = \begin{pmatrix} x_\alpha^2 + x_\alpha'^2 & x'_\alpha y'_\alpha & f_0 x'_\alpha & x_\alpha y_\alpha & 0 & 0 & f_0 x_\alpha & 0 & 0 \\ x'_\alpha y'_\alpha & x_\alpha^2 + y_\alpha'^2 & f_0 y'_\alpha & 0 & x_\alpha y_\alpha & 0 & 0 & f_0 x_\alpha & 0 \\ f_0 x'_\alpha & f_0 y'_\alpha & f_0^2 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_\alpha y_\alpha & 0 & 0 & y_\alpha^2 + x_\alpha'^2 & x'_\alpha y'_\alpha & f_0 x'_\alpha & f_0 y_\alpha & 0 & 0 \\ 0 & x_\alpha y_\alpha & 0 & x'_\alpha y'_\alpha & y_\alpha^2 + y_\alpha'^2 & f_0 y'_\alpha & 0 & f_0 y_\alpha & 0 \\ 0 & 0 & 0 & f_0 x'_\alpha & f_0 y'_\alpha & f_0^2 & 0 & 0 & 0 \\ f_0 x_\alpha & 0 & 0 & f_0 y_\alpha & 0 & 0 & f_0^2 & 0 & 0 \\ 0 & f_0 x_\alpha & 0 & 0 & f_0 y_\alpha & 0 & 0 & f_0^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{4.67}$$

If we regard each $\boldsymbol{\xi}_\alpha$ as perturbed from its true value $\bar{\boldsymbol{\xi}}_\alpha$ by independent Gaussian noise of mean $\boldsymbol{0}$ and covariance matrix $V[\boldsymbol{\xi}_\alpha]$, maximum likelihood (ML) estimation of the fundamental matrix is to minimize the Mahalanobis distance

$$J_{\mathrm{ML}} = \sum_{\alpha=1}^{N} (\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha](\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha)), \tag{4.68}$$

subject to

$$(\boldsymbol{u}, \bar{\boldsymbol{\xi}}_\alpha) = 0. \tag{4.69}$$

We can eliminate the constraint of Eq. (4.69) by using Lagrange multipliers and reduce the problem to unconstrained minimization of

$$J_{\mathrm{ML}} = \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)^2}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}. \tag{4.70}$$

This function can be minimized by various means such as *FNS* (*Fundamental Numerical Scheme*) of Chojnacki et al. (2000), *HEIV* (*Heteroscedastic Errors-In-Variables*) of Leedan and Meer (2000), and projective Gauss-Newton iterations of Sugaya and Kanatani (2007a,c), but all requires iterations, which sometimes do not converge in the presence of large noise. If the denominators in Eq. (4.70) are replaced by a constant, the problem reduces to least squares, minimizing Eq. (4.65). The Taubin method replaces the denominators in Eq. (4.70) by their average and minimizes

$$J_{\rm TB} = \frac{\sum_{\alpha=1}^{N} (\boldsymbol{u}, \boldsymbol{\xi}_\alpha)^2}{\sum_{\alpha=1}^{N} (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})/N} = \frac{N(\boldsymbol{u}, \boldsymbol{M}\boldsymbol{u})}{(\boldsymbol{u}, \boldsymbol{N}\boldsymbol{u})}, \tag{4.71}$$

where $\boldsymbol{M}$ is the matrix in Eq. (4.62) and the matrix $\boldsymbol{N}$ is defined by

$$\boldsymbol{N} = \sum_{\alpha=1}^{N} V_0[\boldsymbol{\xi}_\alpha]. \tag{4.72}$$

Eq. (4.71) is the ratio of quadratic forms, called the *Rayleigh quotient*, and is minimized by the generalized eigenvector of the generalized eigenvalue problem

$$\boldsymbol{M}\boldsymbol{u} = \lambda \boldsymbol{N}\boldsymbol{u}, \tag{4.73}$$

for the smallest generalized eigenvalue. However, this problem cannot be solved by the standard method (Appendix C), because the matrix $V_0[\boldsymbol{\xi}_\alpha]$ in Eq. (4.67) and hence the matrix $\boldsymbol{N}$ in Eq. (4.72) are singular, the elements in the ninth column and the ninth row all being 0. So, we reduce Eq. (4.73) to a generalized eigenvalue problem of a smaller dimension. We decompose the 9-D vectors $\boldsymbol{u}$, $\boldsymbol{\xi}_\alpha$ and the 9×9 matrix $V_0[\boldsymbol{\xi}_\alpha]$ into their 8-D parts and the 9th elements in the form

$$\boldsymbol{u} = \begin{pmatrix} \boldsymbol{v} \\ F_{33} \end{pmatrix}, \quad \boldsymbol{\xi}_\alpha = \begin{pmatrix} \boldsymbol{z}_\alpha \\ f_0^2 \end{pmatrix}, \quad V_0[\boldsymbol{\xi}_\alpha] = \begin{pmatrix} V_0[\boldsymbol{z}_\alpha] & \boldsymbol{0} \\ \boldsymbol{0}^\top & 0 \end{pmatrix}, \tag{4.74}$$

where $\boldsymbol{z}_\alpha$ is the 8-D vector $\boldsymbol{z}_\alpha$ in Eq. (4.5) and $V_0[\boldsymbol{z}_\alpha]$ is the 8×8 matrix in Eq. (4.6). Then, Eq. (4.73) is written as

$$\begin{pmatrix} \sum_{\alpha=1}^{N} \boldsymbol{z}_\alpha \boldsymbol{z}_\alpha^\top & f_0^2 \sum_{\alpha=1}^{N} \boldsymbol{z}_\alpha \\ f_0^2 \sum_{\alpha=1}^{N} \boldsymbol{z}_\alpha^\top & N f_0^4 \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ F_{33} \end{pmatrix} = \lambda \begin{pmatrix} \boldsymbol{N}_{\rm TB} & \boldsymbol{0} \\ \boldsymbol{0}^\top & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{v} \\ F_{33} \end{pmatrix}, \tag{4.75}$$

where the 8×8 matrix $\boldsymbol{N}_{\rm TB}$ is defined by the second of Eqs. (4.8). If we define $\bar{\boldsymbol{z}}_\alpha$ and $\mathrm{o}\tilde{\boldsymbol{z}}_\alpha$ by Eqs. (4.7) and the 8×8 matrix $\boldsymbol{M}_{\rm TB}$ by the first of Eqs. (4.8), we have the identity

$$\sum_{\alpha=1}^{N} \boldsymbol{z}_\alpha \boldsymbol{z}_\alpha^\top = \boldsymbol{M}_{\rm TB} + \bar{\boldsymbol{z}} \bar{\boldsymbol{z}}^\top. \tag{4.76}$$

Hence, Eq. (4.75) splits into the following two equations:

$$\boldsymbol{M}_{\rm TB} \boldsymbol{v} = \lambda \boldsymbol{N}_{\rm TB} \boldsymbol{v}, \qquad (\bar{\boldsymbol{z}}, \boldsymbol{v}) + f_0^2 F_{33} = 0. \tag{4.77}$$

The second equation determines F_{33} in terms of $\boldsymbol{v}$ by $F_{33} = -(\bar{\boldsymbol{z}}, \boldsymbol{v})/f_0^2$, so if we solve the first equation, which is the same as Eq. (4.9), the 9-D vector $\boldsymbol{u}$ (normalized to unit norm) is given by Eq. (4.10). Thus, the procedure in Sec. 4.3.2 is justified.

C. Generalized eigenvalue problem

The generalized eigenvalue problem can be solved by the following procedure:

Input: An $n \times n$ symmetric matrix $\boldsymbol{A}$ and an $n \times n$ positive definite symmetric matrix $\boldsymbol{G}$.

Output: The generalized eigenvalues λ_1, ..., λ_n and the corresponding generalized eigenvectors $\boldsymbol{w}_1$, ..., $\boldsymbol{w}_n$ (normalized to unit norm) that satisfy

$$\boldsymbol{A}\boldsymbol{w}_i = \lambda_i \boldsymbol{G}\boldsymbol{w}_i. \tag{4.78}$$

(1) Compute the eigenvalues μ_1, ..., μ_n (> 0) of $\boldsymbol{G}$ and the corresponding unit eigenvectors $\boldsymbol{g}_1$, ..., $\boldsymbol{g}_n$.
(2) Define the following $n \times n$ matrices $\boldsymbol{T}$ and $\tilde{\boldsymbol{A}}$:

$$\boldsymbol{T} = \frac{\boldsymbol{g}_1 \boldsymbol{g}_1^\top}{\sqrt{\mu_1}} + \cdots + \frac{\boldsymbol{g}_n \boldsymbol{g}_n^\top}{\sqrt{\mu_n}}, \qquad \tilde{\boldsymbol{A}} = \boldsymbol{T}\boldsymbol{A}\boldsymbol{T}. \tag{4.79}$$

(3) Compute the eigenvalues λ_1, ..., λ_n of $\tilde{\boldsymbol{A}}$ and the corresponding unit eigenvectors $\boldsymbol{u}_1$, ..., $\boldsymbol{u}_n$.
(4) Return λ_1, ..., λ_n and the unit vectors $\boldsymbol{w}_1$, ..., $\boldsymbol{w}_n$ defined by

$$\boldsymbol{w}_1 = \mathcal{N}[\boldsymbol{T}\boldsymbol{u}_1], \quad ..., \quad \boldsymbol{w}_n = \mathcal{N}[\boldsymbol{T}\boldsymbol{u}_n]. \tag{4.80}$$

This procedure is justified as follows. Since $\{\boldsymbol{g}_i\}$ are an orthonormal system, the first of Eqs. (4.79) implies

$$\boldsymbol{T}^2 = \frac{\boldsymbol{g}_1 \boldsymbol{g}_1^\top}{\mu_1} + \cdots + \frac{\boldsymbol{g}_n \boldsymbol{g}_n^\top}{\mu_n} = \boldsymbol{G}^{-1}. \tag{4.81}$$

Multiplied by $\boldsymbol{T}$ from left, Eq. (4.78) is written in the form

$$\boldsymbol{T}\boldsymbol{A}\boldsymbol{T}\boldsymbol{T}^{-1}\boldsymbol{w}_i = \lambda_i \boldsymbol{T}\boldsymbol{G}\boldsymbol{T}\boldsymbol{T}^{-1}\boldsymbol{w}_i. \tag{4.82}$$

However, we see that

$$\boldsymbol{T}\boldsymbol{G}\boldsymbol{T} = \boldsymbol{T}(\boldsymbol{T}^2)^{-1}\boldsymbol{T} = \boldsymbol{T}\boldsymbol{T}^{-1}\boldsymbol{T}^{-1}\boldsymbol{T} = \boldsymbol{I}. \tag{4.83}$$

So, if we define

$$\boldsymbol{u}_i = \boldsymbol{T}^{-1}\boldsymbol{w}_i, \tag{4.84}$$

Eq. (4.82) has the following form of the standard eigenvalue problem:

$$\tilde{\boldsymbol{A}}\boldsymbol{u}_i = \lambda_i \boldsymbol{u}_i. \tag{4.85}$$

Hence, if $\tilde{\boldsymbol{A}}$ has eigenvalues λ_i with corresponding unit eigenvectors $\boldsymbol{u}_i$, then λ_i are the generalized eigenvalues of Eq. (4.78), and from Eq. (4.84) the corresponding generalized eigenvectors $\boldsymbol{w}_i$ are given by $\boldsymbol{w}_i = \mathcal{N}[\boldsymbol{T}\boldsymbol{u}_i]$ after normalization.

D. Free focal length method

From Eq. (4.22), we can immediately see that the following $\boldsymbol{e}$ and $\boldsymbol{e}'$ are the eigenvectors of $\boldsymbol{F}^\top$ and $\boldsymbol{F}$, respectively, for eigenvalue 0:

$$\boldsymbol{e} \simeq \mathrm{diag}(1, 1, \frac{f_0}{f})\boldsymbol{t}, \qquad \boldsymbol{e}' \simeq \mathrm{diag}(1, 1, \frac{f_0}{f'})\boldsymbol{R}^\top \boldsymbol{t}. \tag{4.86}$$

Hence, we obtain

$$\boldsymbol{t} \simeq \mathrm{diag}(1, 1, \frac{f}{f_0})\boldsymbol{e} \simeq \boldsymbol{R}\mathrm{diag}(1, 1, \frac{f'}{f_0})\boldsymbol{e}'. \tag{4.87}$$

Substituting this into Eq. (4.22), we obtain

$$\boldsymbol{F} \simeq \boldsymbol{e} \times \mathrm{diag}(1, 1, \frac{f_0}{f})\boldsymbol{R}\mathrm{diag}(1, 1, \frac{f'}{f_0}) \simeq \mathrm{diag}(1, 1, \frac{f}{f_0})\boldsymbol{R}\mathrm{diag}(1, 1, \frac{f_0}{f'}) \times \boldsymbol{e}', \tag{4.88}$$

where we have used the identity $(\boldsymbol{A}\boldsymbol{v}) \times \boldsymbol{I} = (\boldsymbol{A}^{-1})^\top (\boldsymbol{v} \times \boldsymbol{I})\boldsymbol{A}^{-1}$ for an arbitrary vector $\boldsymbol{v}$ and an arbitrary nonsingular matrix $\boldsymbol{A}$. The product $\boldsymbol{A} \times \boldsymbol{v}$ of a 3×3 matrix $\boldsymbol{A}$ and a 3-D vector is defined to be $\boldsymbol{A}(\boldsymbol{v} \times \boldsymbol{I})^\top$. Equation (4.88) implies

$$\begin{aligned} \boldsymbol{F}\mathrm{diag}(1, 1, \frac{f_0}{f'}) &\simeq \boldsymbol{e} \times \mathrm{diag}(1, 1, \frac{f_0}{f})\boldsymbol{R}, \\ \mathrm{diag}(1, 1, \frac{f_0}{f})\boldsymbol{F} &\simeq \boldsymbol{R}\mathrm{diag}(1, 1, \frac{f_0}{f'}) \times \boldsymbol{e}'. \end{aligned} \tag{4.89}$$

Eliminating $\boldsymbol{R}$ by using the orthogonality relation $\boldsymbol{R}^\top \boldsymbol{R} = \boldsymbol{R}\boldsymbol{R}^\top = \boldsymbol{I}$, we obtain the following *Kruppa equations* [Bougnoux (1998), Hartley and Zisserman (2000)]:

$$\begin{aligned} \boldsymbol{F}\mathrm{diag}(1, 1, \frac{f_0^2}{f'^2})\boldsymbol{F}^\top &\simeq \boldsymbol{e} \times \mathrm{diag}(1, 1, \frac{f_0^2}{f^2}) \times \boldsymbol{e}, \\ \boldsymbol{F}^\top \mathrm{diag}(1, 1, \frac{f_0^2}{f^2})\boldsymbol{F} &\simeq \boldsymbol{e}' \times \mathrm{diag}(1, 1, \frac{f_0^2}{f'^2}) \times \boldsymbol{e}'. \end{aligned} \tag{4.90}$$

In terms of $\{\xi, \eta\}$ defined by Eqs. (4.25), these equations are rewritten as

$$\begin{aligned} \boldsymbol{F}(\boldsymbol{I} + \eta \boldsymbol{k}\boldsymbol{k}^\top)\boldsymbol{F}^\top &\simeq \boldsymbol{e} \times (\boldsymbol{I} + \xi \boldsymbol{k}\boldsymbol{k}^\top) \times \boldsymbol{e}, \\ \boldsymbol{F}^\top (\boldsymbol{I} + \xi \boldsymbol{k}\boldsymbol{k}^\top)\boldsymbol{F} &\simeq \boldsymbol{e}' \times (\boldsymbol{I} + \eta \boldsymbol{k}\boldsymbol{k}^\top) \times \boldsymbol{e}', \end{aligned} \tag{4.91}$$

which reduce to

$$\begin{aligned} \boldsymbol{F}\boldsymbol{F}^\top + \eta(\boldsymbol{F}\boldsymbol{k})(\boldsymbol{F}\boldsymbol{k})^\top &\simeq \boldsymbol{P}_{\mathbf{e}} + \xi(\boldsymbol{e} \times \boldsymbol{k})(\boldsymbol{e} \times \boldsymbol{k})^\top, \\ \boldsymbol{F}^\top \boldsymbol{F} + \xi(\boldsymbol{F}^\top \boldsymbol{k})(\boldsymbol{F}^\top \boldsymbol{k})^\top &\simeq \boldsymbol{P}_{\mathbf{e}'} + \eta(\boldsymbol{e}' \times \boldsymbol{k})(\boldsymbol{e}' \times \boldsymbol{k})^\top, \end{aligned} \tag{4.92}$$

where we define the projection matrices

$$\boldsymbol{P}_{\mathbf{e}} = \boldsymbol{I} - \boldsymbol{e}\boldsymbol{e}^\top, \qquad \boldsymbol{P}_{\mathbf{e}'} = \boldsymbol{I} - \boldsymbol{e}'\boldsymbol{e}'^\top, \tag{4.93}$$

onto the planes perpendicular to $\boldsymbol{e}$ and $\boldsymbol{e}'$, respectively. Multiplying $\boldsymbol{k}$ from the right on both sides of Eqs. (4.92), we obtain

$$\boldsymbol{F}\boldsymbol{F}^\top \boldsymbol{k} + \eta(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})\boldsymbol{F}\boldsymbol{k} = c\boldsymbol{P}_{\mathbf{e}}\boldsymbol{k}, \qquad \boldsymbol{F}^\top \boldsymbol{F}\boldsymbol{k} + \xi(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})\boldsymbol{F}^\top \boldsymbol{k} = c'\boldsymbol{P}_{\mathbf{e}'}\boldsymbol{k}, \tag{4.94}$$

where c and c' are unknown constants. Computing the inner product of $\boldsymbol{k}$ and the second of Eqs. (4.94) on both sides and the inner product of $\boldsymbol{F}^\top\boldsymbol{k}$ and the second of Eqs. (4.94), on both sides, we obtain

$$\begin{aligned}\|\boldsymbol{F}\boldsymbol{k}\|^2 + (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2\xi &= c'\|\boldsymbol{e}' \times \boldsymbol{k}\|^2,\\ (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k}) + (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})\|\boldsymbol{F}^\top\boldsymbol{k}\|^2\xi &= c'(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k}),\end{aligned} \tag{4.95}$$

which can be solved for $\{\xi,\ c'\}$, resulting in the first of Eqs. (4.29). Similarly, computing the inner product of $\boldsymbol{k}$ and the first of Eqs. (4.94) on both sides and the inner product of $\boldsymbol{F}\boldsymbol{k}$ and the first of Eqs. (4.94) on both sides, we obtain

$$\begin{aligned}\|\boldsymbol{F}^\top\boldsymbol{k}\|^2 + \eta(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})^2 &= c\|\boldsymbol{e} \times \boldsymbol{k}\|^2,\\ (\boldsymbol{k}, \boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{k}) + \eta(\boldsymbol{k}, \boldsymbol{F}\boldsymbol{k})\|\boldsymbol{F}\boldsymbol{k}\|^2 &= c(\boldsymbol{k}, \boldsymbol{F}^\top\boldsymbol{k}),\end{aligned} \tag{4.96}$$

which can be solved for $\{\eta,\ c\}$, resulting in the second of Eqs. (4.29) (note the identity $(\boldsymbol{k}, \boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{k}) = (\boldsymbol{k}, \boldsymbol{F}\boldsymbol{F}^\top\boldsymbol{F}\boldsymbol{k})$). Our derivation is essentially the same as Bougnoux (1998) but has a slightly different appearance.

E. Equality enforcement

If we equate ξ with η in Eq. (4.31) and ignore the higher order terms $\cdots$, we have

$$K(\xi) = \frac{1}{2}\Big(H_{11}(\xi - \xi_0)^2 + 2H_{12}(\xi - \xi_0)(\xi - \eta_0) + H_{22}(\xi - \eta_0)^2\Big). \tag{4.97}$$

Differentiating this with respect to ξ and setting the result to 0, we have

$$H_{11}(\xi - \xi_0) + H_{12}(\xi - \xi_0) + H_{12}(\xi - \eta_0) + H_{22}(\xi - \eta_0) = 0. \tag{4.98}$$

Solving this for ξ, we obtain ξ and η $(= \xi)$ in the form of Eq. (4.33).

F. Generalized inverse

The *singular value decomposition* (*SVD*) of an $m \times n$ matrix $\boldsymbol{A}$ has the form

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top, \qquad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_1 & & & & \\ & \ddots & & & \\ & & \sigma_r & & \\ & & & 0 & \\ & & & & \ddots \end{pmatrix}, \tag{4.99}$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are $m \times m$ and $n \times n$ orthogonal matrices, respectively, and $\boldsymbol{\Sigma}$ is an $m \times n$ matrix. The numbers $\sigma_1 \geq \sigma_2 \geq \cdots$ are the *singular values*, and the number r $(\leq \min(m, n))$ is the *rank* of $\boldsymbol{A}$. Let $\boldsymbol{\Sigma}^-$ be the $n \times m$ matrix

$$\boldsymbol{\Sigma}^- = \begin{pmatrix} 1/\sigma_1 & & & & \\ & \ddots & & & \\ & & 1/\sigma_r & & \\ & & & 0 & \\ & & & & \ddots \end{pmatrix}. \tag{4.100}$$

The (*Moore-Penrose*) *generalized inverse* (or *pseudoinverse*) of $\boldsymbol{A}$ is the $n \times m$ matrix

$$\boldsymbol{A}^- = \boldsymbol{V}\boldsymbol{\Sigma}^-\boldsymbol{U}^\top. \tag{4.101}$$

In practice, the rank r need to be specified by the user, since a theoretically zero value is treated as a very small nonzero value in floating-point arithmetic. If $r = \min(n, m)$, in particular, we can easily see that

$$\boldsymbol{A}^- = \begin{cases} (\boldsymbol{A}^\top\boldsymbol{A})^{-1} & m \geq n \\ (\boldsymbol{A}\boldsymbol{A}^\top)^{-1} & m \leq n \end{cases}. \tag{4.102}$$

The n-D vector $\boldsymbol{x}$ that minimizes

$$J = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2, \tag{4.103}$$

for an m-D constant vector $\boldsymbol{b}$ is given by

$$\boldsymbol{x} = \boldsymbol{A}^-\boldsymbol{b}. \tag{4.104}$$

This is shown as follows. Since the norm $\|\cdot\|$ is invariant to orthogonal transformations, i.e., $\|\boldsymbol{a}\| = \|\boldsymbol{U}\boldsymbol{a}\|$ for any orthogonal matrix $\boldsymbol{U}$, we see that

$$\begin{aligned} J &= \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|^2 = \|\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top\boldsymbol{x} - \boldsymbol{b}\|^2 = \|\boldsymbol{\Sigma}\boldsymbol{V}^\top\boldsymbol{x} - \boldsymbol{U}^\top\boldsymbol{b}\|^2 \\ &= \|\boldsymbol{\Sigma}\boldsymbol{y} - \boldsymbol{c}\|^2 = \sum_{i=1}^{r}(\sigma_i y_i - c_i)^2 + \sum_{i=r+1}^{m} c_i^2, \end{aligned} \tag{4.105}$$

where we put $\boldsymbol{y} = \boldsymbol{V}^\top\boldsymbol{x}$ and $\boldsymbol{c} = \boldsymbol{U}^\top\boldsymbol{b}$. Thus, J is minimized by $y_1 = c_1/\sigma_1$, ..., $y_r = c_r/\sigma_r$. If $r < n$, then y_{r+1}, ..., y_n are indeterminate, so we let them be all 0, which minimizes $\|\boldsymbol{y}\| = \|\boldsymbol{x}\|$ among all solutions. Thus, we have

$$\boldsymbol{x} = \boldsymbol{V}\boldsymbol{y} = \boldsymbol{V}\boldsymbol{\Sigma}^-\boldsymbol{c} = \boldsymbol{V}\boldsymbol{\Sigma}^-\boldsymbol{U}^\top\boldsymbol{b} = \boldsymbol{A}^-\boldsymbol{b}. \tag{4.106}$$

References

Bartoli, A., Sturm, P. (2004). Nonlinear estimation of fundamental matrix with minimal parameters, *IEEE Trans. Patt. Anal. Mach. Intell.*, **26**, 3, pp. 426–432.

Bougnoux, S. (1998). From projective to Euclidean space under any practical situation, a criticism of self calibration, in *Proc 6th Int. Conf. Comput. Vision (ICCV'98)* (Bombay, India), pp. 790–796.

Brooks, M. J., de Agapito, L., Huynh, D. Q. and Baumela, L. (1998). Towards robust metric reconstruction via a dynamic uncalibrated stereo head. *Image Vision Comput.* **16**, 14, pp. 989–1002.

Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D. (2000). On the fitting of surfaces to data with covariances, *IEEE Trans. Patt. Anal. Mach. Intell.*, **22**, 11, pp. 1294–1303.

Chojnacki, W., Brooks, M.J., van den Hengel, A., Gawley, D. (2004). A new constrained parameter estimator for computer vision applications, *Image Vis. Comput.*, **22**, 2, pp. 85–91.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, *Commun. ACM*, **24**, 6, pp. 381–395.

Harris, C. and Stephens, M. (1988). A combined corner and edge detector, in *Proc. 4th Alvey Vision Conf.* (Manchester, U.K.), pp. 147–151.

Hartley, R. I. (1992). Estimation of relative camera position for uncalibrated cameras. in *Proc. 2nd Euro. Conf. Comput. Vision (ECCV'92)* (Santa Margherita Ligure, Italy), pp. 579–587.

Hartley, R. I. (1997). In defense of the eight-point algorithm, *IEEE Trans. Patt. Anal. Mach. Intell.*, **19**, 6, pp. 580–593.

Hartley, R., and Silpa-Anan, C. (2002). Reconstruction from two views using approximate calibration, in *Proc. 5th Asian Conf. Comput. Vision (ACCV'02)* (Melbourne, Australia), Vol. 1, pp. 338–343.

Hartley, R. and A. Zisserman, A. (2000). *Multiple View Geometry in Computer Vision* (Cambridge University Press, Cambridge).

Huang, T. S. and Faugeras, O. D. (1989). Some properties of the E matrix in two-view motion estimation, *IEEE Trans. Patt. Anal. Mach. Intell.*, **11**, 12, pp. 1310–132.

Kanatani, K. (1993). *Geometric Computation for Machine Vision* (Oxford University Press, Oxford).

Kanatani, K. (1996). *Statistical Optimization for Geometric Computation: Theory and Practice* (Elsevier, Amsterdam), reprinted, 2005 (Dover, New York).

Kanatani, K. (2000). Optimal fundamental matrix computation: Algorithm and reliability analysis, in *Proc. 6th Symp. on Sensing via Image Information (SSII'00)* (Yokohama, Japan), pp. 291–298.

Kanatani, K. (2008). Latest progress of 3-D reconstruction from multiple camera images, in X. P. Guo (ed.), *Robotics Research Trends* (Nova Science, Hauppauge, NY), pp. 33–75.

Kanatani, K. (2009). Unified computation of strict maximum likelihood for geometric fitting, in *Proc. IS&T/SPIE Electronic Imaging: Three-Dimensional Imaging Metrology* (San Jose, CA), Vol. 7239.

Kanatani K. and Matsunaga, C. (2000). Closed-form expression for focal lengths from the fundamental matrix, in *Proc. 4th Asian Conf. Comput. Vision (ACCV'00)* (Taipei, Taiwan), Vol. 1, pp. 128–133.

Kanatani, K., Nakatsuji, A. and Sugaya, Y. (2006). Stabilizing the focal length computation for 3-D reconstruction from two uncalibrated views, *Int. J. Comput. Vision*, **66**, 2, pp. 109–122.

Kanatani, K. and Ohta, N. (2003). Comparing optimal three-dimensional reconstruction for finite motion and optical flow, *J. Electronic Imaging*, **12**, 3, pp.

478–488.

Kanatani, K. and Sugaya, Y. (2007a). High accuracy fundamental matrix computation and its performance evaluation, *IEICE Trans. Inf. Syst*, **E90-D**, 2, pp. 579–585.

Kanatani, K. and Y. Sugaya, Y. (2007b). Extended FNS for constrained parameter estimation, in *Proc. Meeting on Image Recognition and Understanding (MIRU'07)* (Hiroshima, Japan), pp. 219–226.

Kanatani, K. and Sugaya, Y. (2007c). Performance evaluation of iterative geometric fitting algorithms, *Comp. Stat. Data Anal.*, **52**, 2, pp. 1208–1222.

Kanatani, K. and Sugaya, Y. (2009). Compact fundamental matrix computation, in *Proc. IEEE Pacific-Rim Symp. on Image and Video Technology (PSIVT'09)* (Tokyo, Japan), pp. 179–190.

Kanatani, K., Sugaya, Y. and Niitsuma, H. (2008). Triangulation from two views revised: Hartley-Sturm vs. optimal correction, in *Proc. 19th British Machine Vision Conf. (BMVC'08)* (Leeds, U.K.), pp. 173–182.

Kanazawa, Y. and Kanatani, K. (2004). Robust image matching preserving global consistency, in *Proc. 6th Asian Conf. Comput. Vision (ACCV'04)* (Jeju, Korea), Vol. 2, pp. 1128–1133.

Leedan, Y. and Meer, P. (2000). Heteroscedastic regression in computer vision: Problems with bilinear constraint, *Int. J. Comput. Vision* **37**, 2, pp.127–150.

Maybank, S. (1990). The projective geometry of ambiguous surface, *Phil. Trans. Roy. Soc. Lond*, A**332**, pp. 1-47.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints, *Int. J. Comput. Vision.*, **60**, 2, pp. 91–110.

Matas, J., Chum, O., Urban, M. and T Pajdla, T. (2002). Robust wide baseline stereo from maximally stable extremal regions, in *Proc. 13th British Machine Vision Conf. (BMVC'02)* (Cardiff. U.K.), pp. 384–393.

Maybank, S. (1991). Ambiguity in reconstruction from image correspondences *Image Vision Comput.*, **9**, 2, pp. 93-99.

Newsam, G. N., Huynh, D. Q., Brooks, M. J., and Pan, H.-P. (1996), Recovering unknown focal lengths in self-calibration: An essentially linear algorithm and degenerate configurations, *Int. Arch. Photogram. Remote Sensing*, **31**, B3-III, pp. 575–580.

Pan, H.-P., Brooks, M. J., and Newsam, G. (1995a). Image resituation: initial theory. in *Proc. SPIE: Videometrics IV* (Philadelphia, PA), pp 162–173.

Pan, H.-P., Huynh, D. Q., and Hamlyn, G. (1995b). Two-image resituation: Practical algorithm. in *Proc. SPIE: Videometrics IV*, (Philadelphia, PA), pp. 174–190.

Press, W. H, Teukolsky, S. A., Vettering, W. T. Flannery, B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed. (Cambridge University Press, Cambridge).

Rousseeuw, P. J. and Leroy, A. M. (1987). *Robust Regression and Outlier Detection*

(Wiley, New York).

Smith, S. M. and Brady, J. M. (1997). SUSAN—A new approach to low level image processing, *Int. J. Comput. Vision*, **23** 1, pp. 45–78.

Sturm, P. (2001). On focal length calibration from two views, in *Proc. IEEE Conf. Comput. Vision Pattern Recog.* (Kauai, HI), Vol. 2, pp. 145–150.

Taubin, G. (1991). Estimation of planar curves, surfaces, and non-planar space curves defined by implicit equations with applications to edge and rage image segmentation, *IEEE Trans. Patt. Anal. Mach. Intell.*, **13**, 11, pp. 1115–1138.

Tomasi, C. and Kanade, K. (1991). Detection and tracking of point features CMU Tech. Rep. CMU-CS-91-132, Carnegie Mellon University, Pittsburgh, PA. http://vision.stanford.edu/~birch/klt/

Triggs, B., McLauchlan, P. F., Hartley, R. I. and Fitzgibbon, A. (2000). Bundle adjustment—A modern synthesis, in B. Triggs, A. Zisserman, and R. Szeliski, (eds.), *Vision Algorithms: Theory and Practice* (Springer, Berlin), pp. 298–375.

Zhang, Z., Deriche, R., Faugeras, O. and Luong, Q.-T. (1995). A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry, *Artif. Intell.*, **78**, 1/2, pp.87–119.