*Research Paper*

Optimal Two-View Planar Scene Triangulation

Kenichi Kanatani[†1] and Hirotaka Niitsuma[†1]

We present a new algorithm for optimally computing from point correspondences over two images their 3-D positions using the knowledge that they are constrained to be on a planar surface. We consider two cases: the case in which the plane and camera parameters are known and the case in which they are not. In the former, we show how observed point correspondences are optimally corrected so that they are compatible with the homography between the two images. In the latter, we show how the homography is optimally estimated by iteratively using the triangulation procedure. Although the accuracy improvement over existing methods is very small, our algorithm has a theoretical merit of computing an exact maximum likelihood solution.

1. Introduction

Computing the 3-D position of a point from its projection in two images is called *triangulation* and is a fundamental tool of computer vision[4]. The basic principle is to compute the intersection of the rays, or the lines of sight, starting from the camera lens center, or the viewpoint, and passing through the corresponding image points. However, point correspondence detection using an image processing operation incurs errors to some extent, and the two rays may not intersect. A naive solution is to compute the midpoint of the shortest segment connecting the two rays (**Fig. 1** (a)), but Kanatani[8] and Hartley and Sturm[5] pointed out that for optimal estimation the corresponding points should be displaced so that the rays meet in the scene (Fig. 1 (b)) in such a way that the sum of the square displaced distances, or the *reprojection error*, is minimized. For this, Hartley and Sturm[5] presented an algorithm that reduces to solving a 6th degree polynomial, while Kanazawa and Kanatani[14] gave a first approximation in an analytical form. Later, Kanatani et al.[12] showed that the first approximation is sufficiently accurate and that a few iterations lead to complete agreement with the Hartley-

†1 Okayama University
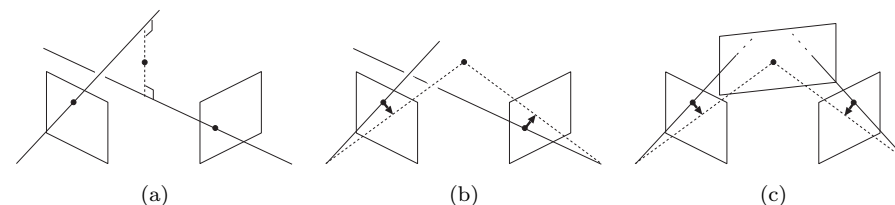
Fig. 1 Triangulation. (a) The midpoint of the shortest segment connecting the rays. (b) The points are optimally corrected so that their rays intersect. (c) The points are optimally corrected so that their rays intersect on the plane.

Sturm solution with far more efficiency. Lindstrom[15] and Tossavainen[19] further improved this approach.

The aim of this paper is to demonstrate that exactly the same holds when the points we are viewing are constrained to be on a planar surface. This is a common situation in indoor and urban scenes. If a 3-D point is constrained to be on a known plane, the corresponding points must be displaced so that the rays not merely intersect but also meet on that plane (Fig. 1 (c)). We call this *planar triangulation* after Chum et al.[3]. A first approximation solution was given by Kanazawa and Kanatani[13], while Chum et al.[3] presented an algorithm that reduces to solving an 8th degree polynomial. The purpose of this paper is to demonstrate that a few iterations of the first approximation lead to an optimal solution. We consider two cases:

- The plane and camera parameters are known.
- The plane and camera parameters are not known.

The algorithm of Chum et al.[3] deals with the former. The latter case could be solved using the bundle adjustment approach, as done by Bartoli and Sturm[1] for an arbitrary number of images. Here, we present a much simpler method to deal with the latter case by extending the former. We highlight the fact that our optimal triangulation procedure for the former case can straightforwardly be extended to the latter.

In Section 2, we summarize the fundamentals about planar projection and homographies. In Sections 3 and 4, we present an iterative algorithm for optimal planar triangulation for known plane and camera parameters. In Section 5, we show that our optimal triangulation procedure can automatically be extended to

the case of unknown plane and camera parameters. In Section 6, we do numerical simulations and demonstrate that our algorithm and the first approximation of Kanazawa and Kanatani [13] give practically the same value.
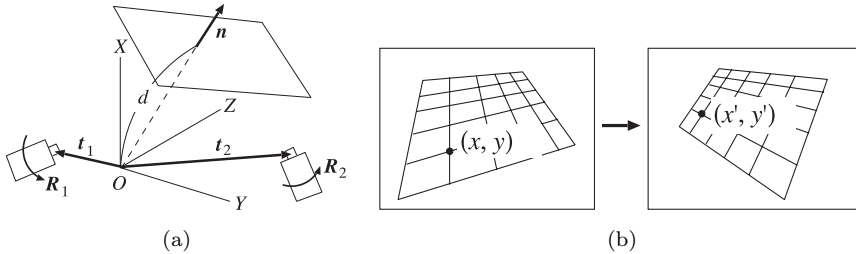
2. Planar Surface and Homography

First, we summarize the relationship between planar surfaces and homographies. Consider a plane with a unit surface normal $\boldsymbol{n}$ at distance d from the origin of an XYZ world coordinate system (**Fig. 2** (a)). We take images of this plane from two positions. The ith camera, $i = 1$, 2, is translated from the world origin O by $\boldsymbol{t}_i$ after rotated by $\boldsymbol{R}_i$. We call $\{\boldsymbol{t}_i, \boldsymbol{R}_i\}$ the *motion parameters* of the ith camera. We assume that by prior camera calibration the image origin is placed at the principal point and that the aspect ratio is 1 with no image skew.

The image of the plane taken from the first position, let us call it the "first image," and the image taken from the second position, let us call it the "second image," are related by the *homography* [4],[8] (Fig. 2 (b))

$$x' = f_0 \frac{h_{11}x + h_{12}y + h_{13}f_0}{h_{31}x + h_{32}y + h_{33}f_0}, \qquad y' = f_0 \frac{h_{21}x + h_{22}y + h_{23}f_0}{h_{31}x + h_{32}y + h_{33}f_0}, \qquad (1)$$

where f_0 is a scale factor of approximately the size of the image for stabilizing numerical computation with finite length. The 3×3 matrix $\boldsymbol{H} = (h_{ij})$ is determined by the parameters $\{\boldsymbol{n}, d\}$ of the plane, the motion parameters $\{\boldsymbol{R}_i, \boldsymbol{t}_i\}$ and the focal lengths f_i, $i = 1$, 2, in the form [4],[8]

$$\boldsymbol{H} = \mathrm{diag}\left(1, 1, \frac{f_0}{f_2}\right)\boldsymbol{R}_2^\top\left(\boldsymbol{I} - \frac{\boldsymbol{t}_2\boldsymbol{n}^\top}{d}\right)\left(\boldsymbol{I} + \frac{\boldsymbol{t}_1\boldsymbol{n}^\top}{d-(\boldsymbol{t}_1,\boldsymbol{n})}\right)\boldsymbol{R}_1\mathrm{diag}\left(1, 1, \frac{f_1}{f_0}\right), \quad (2)$$

Fig. 2   (a) Plane and camera configuration. (b) Two images of the same planar surface are related by a homography.

where $\boldsymbol{I}$ is the unit matrix, and $\mathrm{diag}(a, b, c)$ denotes the diagonal matrix with diagonal elements a, b, and c in that order. Throughout this paper, we denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $(\boldsymbol{a}, \boldsymbol{b})$.

3. Triangulation for Known Plane and Cameras

We first consider the case in which we know $\{\boldsymbol{n}, d\}$, $\{\boldsymbol{R}_i, \boldsymbol{t}_i\}$, and f_i, $i = 1$, 2, hence the homography $\boldsymbol{H}$. In homogeneous coordinates, Eqs. (1) are written as [4],[8]

$$\begin{pmatrix} x'/f_0 \\ y'/f_0 \\ 1 \end{pmatrix} \cong \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix}. \qquad (3)$$

The symbol $\cong$ means equality up to a nonzero constant. Equation (3) can equivalently be written as

$$\begin{pmatrix} x'/f_0 \\ y'/f_0 \\ 1 \end{pmatrix} \times \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \qquad (4)$$

The three components of this equation multiplied by f_0^2 are

$$(\boldsymbol{\xi}^{(1)}, \boldsymbol{h}) = 0, \qquad (\boldsymbol{\xi}^{(2)}, \boldsymbol{h}) = 0, \qquad (\boldsymbol{\xi}^{(3)}, \boldsymbol{h}) = 0, \qquad (5)$$

where we define the 9-D vectors $\boldsymbol{h}$, $\boldsymbol{\xi}^{(1)}$, $\boldsymbol{\xi}^{(2)}$, and $\boldsymbol{\xi}^{(3)}$ by

$$\boldsymbol{h} = (h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32}, h_{33})^\top, \qquad (6)$$

$$\boldsymbol{\xi}^{(1)} = (0, 0, 0, -f_0 x, -f_0 y, -f_0^2, xy', yy', f_0 y')^\top,$$
$$\boldsymbol{\xi}^{(2)} = (f_0 x, f_0 y, f_0^2, 0, 0, 0, -xx', -yx', -f_0 x')^\top,$$
$$\boldsymbol{\xi}^{(3)} = (-xy', -yy', -f_0 y', xx', yx', f_0 x', 0, 0, 0)^\top. \qquad (7)$$

A corresponding point pair (x, y) and (x', y') can be identified with a point $\boldsymbol{p} = (x, y, x', y')^\top$ in the 4-D $xyx'y'$ joint space. Each of the three equations in Eqs. (5) is a second degree polynomial in x, y, x', and y and linear in h_{ij}, defining a hypersurface in this 4-D joint space. However, the identity $x'\boldsymbol{\xi}^{(1)} + y'\boldsymbol{\xi}^{(2)} + f_0\boldsymbol{\xi}^{(3)} = \boldsymbol{0}$ holds, so Eqs. (5) define a 2-D variety, or algebraic manifold, $\mathcal{S}$ in the 4-D joint space.
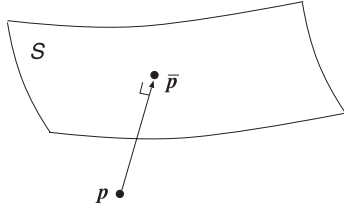
Fig. 3 The point $\boldsymbol{p}$ is orthogonally projected on to $\bar{\boldsymbol{p}}$ on $\mathcal{S}$ in the 4-D joint space.

In the presence of noise, by which we mean uncertainty of image processing operations, the point $\boldsymbol{p}$ is not necessarily on $\mathcal{S}$. Optimal planar triangulation is to displace $\boldsymbol{p}$ to a point $\bar{\boldsymbol{p}}$ on $\mathcal{S}$ in such a way that the *reprojection error*

$$E = (x - \bar{x})^2 + (y - \bar{x})^2 + (x' - \bar{x}')^2 + (y - \bar{y}')^2 = \|\boldsymbol{p} - \bar{\boldsymbol{p}}\|^2, \qquad (8)$$

is minimized subject to the constraint that $(\bar{x}, \bar{y})$ and $(\bar{x}', \bar{y}')$ satisfy the given homography, i.e.,

$$(\boldsymbol{\xi}^{(k)}(\bar{\boldsymbol{p}}), \boldsymbol{h}) = 0, \qquad k = 1, 2, 3. \qquad (9)$$

Geometrically, this means *orthogonally projecting* $\boldsymbol{p}$ onto the variety $\mathcal{S}$ in the 4-D joint space (**Fig. 3**). This can also be viewed as maximum likelihood estimation under isotropic Gaussian noise.

Once such a $\bar{\boldsymbol{p}} = (\bar{x}, \bar{y}, \bar{x}', \bar{y}')^\top$ is obtained, the corresponding 3-D position (X, Y, Z) is determined by solving

$$\begin{pmatrix} \bar{x}/f_0 \\ \bar{y}/f_0 \\ 1 \end{pmatrix} \cong \boldsymbol{P}_1 \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \qquad \begin{pmatrix} \bar{x}'/f_0 \\ \bar{y}'/f_0 \\ 1 \end{pmatrix} \cong \boldsymbol{P}_2 \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}, \qquad (10)$$

where $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ are the 3×4 projection matrices defined as follows [4),12)]:

$$\boldsymbol{P}_1 = \mathrm{diag}\left(1, 1, \frac{f_0}{f_1}\right) \boldsymbol{R}_1^\top (\ \boldsymbol{I} \quad -\boldsymbol{t}_1\ ),$$

$$\boldsymbol{P}_2 = \mathrm{diag}\left(1, 1, \frac{f_0}{f_2}\right) \boldsymbol{R}_2^\top (\ \boldsymbol{I} \quad -\boldsymbol{t}_1\ ). \qquad (11)$$

Equations (10) give four linear equations in X, Y, and Z, but because Eq. (9) is satisfied, a unique solution is obtained [12)].

4. Optimal Planar Triangulation

We now present a new procedure for minimizing Eq. (8) subject to Eq. (9). Since the structure of the problem is very simple, many approaches are conceivable. For example, we may use Eq. (9) to eliminate $\bar{x}'$ and $\bar{y}'$ from E in Eq. (8) and minimize E with respect to $\bar{x}$ and $\bar{y}$ by gradient descent or Newton-like iterations. Here, we take an approach that can be straightforwardly extended to the case of unknown plane and camera parameters, as we show later.

While unconstrained triangulation [12)] involves a single constraint describing the epipolar geometry, planar triangulation is constrained by the three equations in Eq. (9), which are not mutually algebraically independent. For this, the first approximation has already been presented by Kanazawa and Kanatani [13)]. We modify their method so that an optimal solution is obtained by iterations. The procedure is as follows (see Appendix A.1 for the derivation):

(1) Let $E_0 = \infty$ (a sufficiently large number), and define the 4-D vectors

$$\boldsymbol{p} = \begin{pmatrix} x \\ y \\ x' \\ y' \end{pmatrix}, \qquad \hat{\boldsymbol{p}} = \begin{pmatrix} \hat{x} \\ \hat{y} \\ \hat{x}' \\ \hat{y}' \end{pmatrix}, \qquad \tilde{\boldsymbol{p}} = \begin{pmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{x}' \\ \tilde{y}' \end{pmatrix}, \qquad (12)$$

where we let $\hat{x} = x$, $\hat{y} = y$, $\hat{x}' = x'$, $\hat{y}' = y'$, and $\tilde{x} = \tilde{y} = \tilde{x}' = \tilde{y}' = 0$.

(2) Compute the following 9×4 matrices $\boldsymbol{T}^{(1)}$, $\boldsymbol{T}^{(2)}$, and $\boldsymbol{T}^{(3)}$:

$$\boldsymbol{T}^{(1)} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -f_0 & 0 & 0 & 0 \\ 0 & -f_0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hat{y}' & 0 & 0 & \hat{x} \\ 0 & \hat{y}' & 0 & \hat{y} \\ 0 & 0 & 0 & f_0 \end{pmatrix}, \quad \boldsymbol{T}^{(2)} = \begin{pmatrix} f_0 & 0 & 0 & 0 \\ 0 & f_0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\hat{x}' & 0 & -\hat{x} & 0 \\ 0 & -\hat{x}' & -\hat{y} & 0 \\ 0 & 0 & -f_0 & 0 \end{pmatrix},$$

$$T^{(3)} = \begin{pmatrix} -\hat{y}' & 0 & 0 & -\hat{x} \\ 0 & -\hat{y}' & 0 & -\hat{y} \\ 0 & 0 & 0 & -f_0 \\ \hat{x}' & 0 & \hat{x} & 0 \\ 0 & \hat{x}' & \hat{y} & 0 \\ 0 & 0 & f_0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}. \tag{13}$$

(3) Compute the following $\boldsymbol{\xi}^{(1)*}$, $\boldsymbol{\xi}^{(2)*}$, and $\boldsymbol{\xi}^{(3)*}$:

$$\boldsymbol{\xi}^{(1)*} = (0, 0, 0, -f_0\hat{x}, -f_0\hat{y}, -f_0^2, \hat{x}\hat{y}', \hat{y}\hat{y}', f_0\hat{y}')^\top + \boldsymbol{T}^{(1)}\tilde{\boldsymbol{p}},$$
$$\boldsymbol{\xi}^{(2)*} = (f_0\hat{x}, f_0\hat{y}, f_0^2, 0, 0, 0, -\hat{x}\hat{x}', -\hat{y}\hat{x}', -f_0\hat{x}')^\top + \boldsymbol{T}^{(2)}\tilde{\boldsymbol{p}},$$
$$\boldsymbol{\xi}^{(3)*} = (-\hat{x}\hat{y}', -\hat{y}\hat{y}', -f_0\hat{y}', \hat{x}\hat{x}', \hat{y}\hat{x}', f_0\hat{x}', 0, 0, 0)^\top + \boldsymbol{T}^{(3)}\tilde{\boldsymbol{p}}.$$
$$\tag{14}$$

(4) Compute the following 9×9 matrices $V_0^{(kl)}[\boldsymbol{\xi}]$:

$$V_0^{(kl)}[\boldsymbol{\xi}] = \boldsymbol{T}^{(k)}\boldsymbol{T}^{(l)\top}. \tag{15}$$

(5) Compute the 3×3 matrix $\boldsymbol{W} = (W^{(kl)})$

$$\boldsymbol{W} = \begin{pmatrix} (\boldsymbol{h}, V_0^{(11)}[\boldsymbol{\xi}]\boldsymbol{h}) & (\boldsymbol{h}, V_0^{(12)}[\boldsymbol{\xi}]\boldsymbol{h}) & (\boldsymbol{h}, V_0^{(13)}[\boldsymbol{\xi}]\boldsymbol{h}) \\ (\boldsymbol{h}, V_0^{(21)}[\boldsymbol{\xi}]\boldsymbol{h}) & (\boldsymbol{h}, V_0^{(22)}[\boldsymbol{\xi}]\boldsymbol{h}) & (\boldsymbol{h}, V_0^{(23)}[\boldsymbol{\xi}]\boldsymbol{h}) \\ (\boldsymbol{h}, V_0^{(31)}[\boldsymbol{\xi}]\boldsymbol{h}) & (\boldsymbol{h}, V_0^{(32)}[\boldsymbol{\xi}]\boldsymbol{h}) & (\boldsymbol{h}, V_0^{(33)}[\boldsymbol{\xi}]\boldsymbol{h}) \end{pmatrix}_2^-, \tag{16}$$

where $(\,\cdot\,)_r^-$ denotes pseudoinverse with truncated rank r (the smallest eigenvalue is replaced by 0 in its spectral decomposition).

(6) Update $\tilde{\boldsymbol{p}}$ and $\hat{\boldsymbol{p}}$ as follows:

$$\tilde{\boldsymbol{p}} = \sum_{k,l=1}^3 W^{(kl)}(\boldsymbol{\xi}^{(l)*}, \boldsymbol{h})\boldsymbol{T}^{(k)\top}\boldsymbol{h}, \qquad \hat{\boldsymbol{p}} \leftarrow \boldsymbol{p} - \tilde{\boldsymbol{p}}. \tag{17}$$

(7) Evaluate the reprojection error $E = \|\tilde{\boldsymbol{p}}\|^2$. If $E \approx E_0$, return $\hat{\boldsymbol{p}}$ as $\bar{\boldsymbol{p}}$ and stop. Else, let $E_0 \leftarrow E$ and go back to Step (2).

The use of the pseudoinverse with truncated rank 2 in Eq. (16) reflects the fact that only two of the three constraints in Eqs. (5) are algebraically independent. This algorithm computes the same solution as that of Chum et al. [3], which solves an 8th degree polynomial. However, our algorithm involves only a few iterations of linear calculus without requiring any polynomial solver, which is sometimes inefficient and numerically unstable.

Strictly speaking, our procedure is iterative local search, which, theoretically, may not find a true solution, while the algorithm of Chum et al. [3] computes all the roots of an 8th degree polynomial, from among which a true solution is selected. However, we need not worry about false solutions in practice, because the true solution is the *closest* to the observation by the definition of the reprojection error E. In other words, if local minima of E do exist, which is very questionable, they are necessarily located *farther away from the observation*, and hence the solution found by local search should be the true solution except in a vary pathological case, which is difficult to imagine. So, we do not consider such a possibility. The same argument is done for unconstraint triangulation [12].

5. Triangulation for Unknown Plane and Cameras

Next, we consider the case in which $\{\boldsymbol{n}, d\}$, $\{\boldsymbol{R}_i, \boldsymbol{t}_i\}$, $i = 1$, 2, are unknown. As is well known [4],[8], these parameters can be estimated by computing the homography $\boldsymbol{H}$ between the two images, provided that the focal lengths f_i are known; we assume that they are given by prior camera calibration. Once the homography $\boldsymbol{H}$ is known, the plane and camera parameters can be analytically computed [6],[16],[20]–[22] (see Appendix A.2). Thus, the problem reduces to computing the homography $\boldsymbol{H}$ from point correspondences over the two images. The simplest method is to minimize the algebraic distance, which is known by many names such as *least squares* and *DLT (Direct Linear Transformation)* [4], but the accuracy is low in the presence of noise [17]. A method known to be very accurate is what is called *Sampson error minimization* [4], and an iterative scheme was presented by Scoleri et al. [18]. However, Sampson error minimization does not necessarily compute an exactly optimal solution in the sense of maximum likelihood [10]. We now show that by combining the Sampson error minimization with the optimal planar triangulation described in Section 4, we can obtain an exactly optimal $\boldsymbol{H}$. This is the second contribution of this paper.

Our method is based on the theory of Kanatani and Sugaya [10], but the procedure described there applies only to a single constraint, which is the case in ellipse fitting and fundamental matrix computation. Here, we extend their theory to homographies constrained by multiple equations. Given N corresponding points (x_α, y_α) and (x'_α, y'_α), or 4-D joint points $\boldsymbol{p}_\alpha = (x_\alpha, y_\alpha, x'_\alpha, y'_\alpha)^\top$, $\alpha = 1, \ldots, N$, our task is to optimally compute the 9-D vector $\boldsymbol{h}$ that encodes the homography $\boldsymbol{H}$ by minimizing the sum of the reprojection error over all these pairs

$$E = \frac{1}{2} \sum_{\alpha=1}^{N} \left((x_\alpha - \bar{x}_\alpha)^2 + (y_\alpha - \bar{x}_\alpha)^2 + (x'_\alpha - \bar{x}'_\alpha)^2 + (y_\alpha - \bar{y}'_\alpha) \right)^2$$

$$= \frac{1}{2} \sum_{\alpha=1}^{N} \|\boldsymbol{p}_\alpha - \bar{\boldsymbol{p}}_\alpha\|^2, \tag{18}$$

for the true values $\bar{\boldsymbol{p}}_\alpha = (\bar{x}_\alpha, \bar{y}_\alpha, \bar{x}'_\alpha, \bar{y}'_\alpha)^\top$ of $\boldsymbol{p}_\alpha$ subject to the constaint that

$$(\boldsymbol{\xi}^{(k)}(\bar{\boldsymbol{p}}_\alpha), \boldsymbol{h}) = 0, \qquad k = 1, 2, 3, \quad \alpha = 1, \ldots, N. \tag{19}$$

This can also be viewed as maximum likelihood estimation under homogeneous isotropic Gaussian noise.

If the points we are observing are precisely coplanar in the scene and their projections are precisely detected on the image plane, there is a solution of $\bar{\boldsymbol{p}}_\alpha$ and $\boldsymbol{h}$ that makes the reprojection error E in Eq. (18) exactly 0. Here, we are assuming that

- the points we are observing may not precisely be coplanar in the scene, and/or
- their projections on the image plane may not precisely be detected because of uncertainty and inaccuracy of image processing operations.

Whichever is the case, we compute $\bar{\boldsymbol{p}}_\alpha$ and $\boldsymbol{h}$ that minimize Eq. (18). Hence, this method can also be applied to not exactly planar surfaces such as building facades and road surfaces. The procedure for minimizing Eq. (18) is as follows (see Appendix A.3 for the derivation):

(1) Let $E_0 = \infty$ (a sufficiently large number), and give an initial guess of $\boldsymbol{h}$ using any method, say least squares.

(2) Let $\boldsymbol{p}_\alpha$, $\hat{\boldsymbol{p}}_\alpha$, and $\tilde{\boldsymbol{p}}_\alpha$ be the vectors in Eqs. (12) for the αth point $\boldsymbol{p}_\alpha$, $\alpha = 1, \ldots, N$.

(3) Let $\boldsymbol{T}_\alpha^{(1)}$, $\boldsymbol{T}_\alpha^{(2)}$, and $\boldsymbol{T}_\alpha^{(3)}$ be, respectively, the values of $\boldsymbol{T}^{(1)}$, $\boldsymbol{T}^{(2)}$, and $\boldsymbol{T}^{(3)}$

in Eqs. (13) for the αth point $\boldsymbol{p}_\alpha$, $\alpha = 1, \ldots, N$.

(4) Compute the 9×9 matrices $V_0^{(kl)}[\boldsymbol{\xi}_\alpha] = \boldsymbol{T}_\alpha^{(k)} \boldsymbol{T}_\alpha^{(l)\top}$ and the 3×3 matrices $\boldsymbol{W}_\alpha = (W_\alpha^{(kl)})$ in Eq. (16) for the αth point $\boldsymbol{p}_\alpha$, and let the 9-D vectors $\boldsymbol{\xi}_\alpha^{(1)*}$, $\boldsymbol{\xi}_\alpha^{(2)*}$, and $\boldsymbol{\xi}_\alpha^{(3)*}$ be the values of $\boldsymbol{\xi}^{(1)*}$, $\boldsymbol{\xi}^{(2)*}$, and $\boldsymbol{\xi}^{(3)*}$ in Eqs. (14) for the αth point $\boldsymbol{p}_\alpha$, $\alpha = 1, \ldots, N$.

(5) Compute the 9-D unit vector $\boldsymbol{h}$ that minimizes

$$J = \frac{1}{N} \sum_{\alpha=1}^{N} \sum_{k,l=1}^{3} W_\alpha^{(kl)} (\boldsymbol{\xi}_\alpha^{(k)*}, \boldsymbol{h})(\boldsymbol{\xi}_\alpha^{(l)*}, \boldsymbol{h}). \tag{20}$$

(6) Update $\tilde{\boldsymbol{p}}_\alpha$ and $\hat{\boldsymbol{p}}_\alpha$, $\alpha = 1, \ldots, N$, as follows:

$$\tilde{\boldsymbol{p}}_\alpha = \sum_{k,l=1}^{3} W_\alpha^{(kl)} (\boldsymbol{\xi}_\alpha^{(l)*}, \boldsymbol{h}) \boldsymbol{T}_\alpha^{(k)\top} \boldsymbol{h}, \qquad \hat{\boldsymbol{p}}_\alpha \leftarrow \boldsymbol{p}_\alpha - \tilde{\boldsymbol{p}}_\alpha. \tag{21}$$

(7) Evaluate the reprojection error $E = \sum_{\alpha=1}^{N} \|\tilde{\boldsymbol{p}}_\alpha\|^2$. If $E \approx E_0$, return $\boldsymbol{h}$ and stop. Else, let $E_0 \leftarrow E$ and go back to Step (3).

This procedure is identical to the optimal planar triangulation in Section 4 except for Step (5). Equation (20) coincides with what is known as the *Sampson error* [4] if $\boldsymbol{\xi}_\alpha^{(k)*}$ and $\boldsymbol{\xi}_\alpha^{(l)*}$ on the right-hand side are respectively replaced by $\boldsymbol{\xi}_\alpha^{(k)}$ and $\boldsymbol{\xi}_\alpha^{(l)}$, i.e., the values of Eqs. (7) for the αth point $\boldsymbol{p}_\alpha$ (see Appendix A.3 for the meaning of Eq. (20)). It can be minimized by the scheme of Scoleri et al. [18], but here we use a much simpler reformulation of Kanatani et al. [17], which is a direct extension of the *FNS* (*Fundamental Numerical Scheme*) of Chojnacki et al. [2]. The procedure goes as follows (see Appendix A.4 for the derivation):

(1) Provide an initial value $\boldsymbol{h}_0$ for $\boldsymbol{h}$, e.g., by least squares.

(2) Compute the matrices $\boldsymbol{M}$ and $\boldsymbol{L}$ by

$$\boldsymbol{M} = \frac{1}{N} \sum_{\alpha=1}^{N} W_\alpha^{(kl)} \boldsymbol{\xi}_\alpha^{(k)*} \boldsymbol{\xi}_\alpha^{(l)*\top}, \qquad \boldsymbol{L} = \frac{1}{N} \sum_{\alpha=1}^{N} \sum_{k,l=1}^{N} v_\alpha^{(k)} v_\alpha^{(l)} V_0^{(kl)}[\boldsymbol{\xi}_\alpha],$$

$$\tag{22}$$

where we define

$$v_\alpha^{(k)} = \sum_{l=1}^{3} W_\alpha^{(kl)} (\boldsymbol{\xi}_\alpha^{(l)*}, \boldsymbol{h}). \tag{23}$$

(3)   Solve the eigenvalue problem

$$(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{h} = \lambda\boldsymbol{h}, \tag{24}$$

and compute the unit eigenvector $\boldsymbol{h}$ for the smallest eigenvalue λ.

(4)   If $\boldsymbol{h} \approx \boldsymbol{h}_0$, return $\boldsymbol{h}$ and stop. Else, let $\boldsymbol{h}_0 \leftarrow \boldsymbol{h}$, and go back to Step (2).

For unconstrained triangulation, the optimal algorithm of Kanatani et al. [12], which assumes a given fundamental matrix, can automatically be converted to optimal fundamental matrix computation merely by inserting a Sampson error minimization step, as shown by Kanatani and Sugaya [9]. In contrast, the polynomial solving algorithm of Hartley and Sturm [5] cannot be so easily converted to optimal fundamental matrix computation. The situation is the same for planar triangulation. The optimal planar triangulation we proposed in Section 4, which assumes a given homography, can automatically be converted to optimal homography computation merely by inserting a Sampson error minimization step. In contrast, the polynomial solving algorithm of Chum et al. [3] cannot be so easily converted to optimal homography computation.

6. Experiments

Figure 4 (a) shows two images of a simulated grid. The image size is 500×500 pixels; the focal lengths are $f_1 = f_2 = 600$ pixels. We added Gaussian noise of mean 0 and standard deviation 1 pixel to the x and y coordinates of each of the N (= 121) grid points. Then, we reconstructed the 3-D position of each grid point by unconstrained triangulation [12] and by our planar triangulation. Fig. 4 (b) shows the 3-D positions of the grid points. We can see that by planar triangulation (in blue) all the points are on the specified plane but not by unconstrained triangulation (in red). For quantitative evaluation, we measured the root mean square reprojection error

$$e = \sqrt{\frac{1}{N}\sum_{\alpha=1}^{N}\left((\hat{x}_\alpha - x_\alpha)^2 + (\hat{y}_\alpha - y_\alpha)^2 + (\hat{x}'_\alpha - x'_\alpha)^2 + (\hat{y}'_\alpha - y'_\alpha)^2\right)}, \tag{25}$$

where $(\hat{x}_\alpha, \hat{y}_\alpha)$ and $(\hat{x}'_\alpha, \hat{y}'_\alpha)$ are the corrected positions of the observations (x_α, y_α) and (x'_α, y'_α), respectively. We also evaluated the 3-D reconstruction error


(a)   (b)
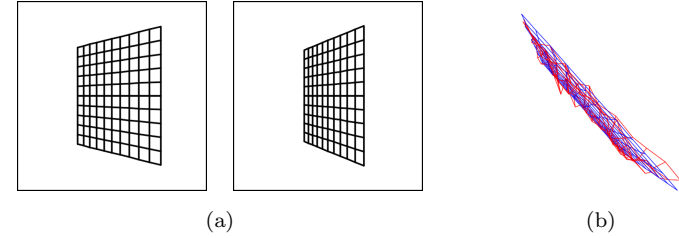
	reprojection error	theoretical expectation	3-D error
unconstrained	1.99503	2.00000	4.24466
planar	1st approx. 2.83127	—	0.95937
	exact 2.83124	2.82843	0.95935

Fig. 4   (a) Simulated images of a planar grid taken from different places. (b) 3-D position of the reconstructed grid. Points reconstructed by planar triangulation (in blue) are on the specified plane, but those reconstructed by unconstrained triangulation [12] (in red) are not necessarily on it. The table below lists the reprojection error, its theoretical expectation, and the average 3-D reconstruction error.

$$D = \sqrt{\frac{1}{N}\sum_{\alpha=1}^{N}\|\hat{\boldsymbol{r}}_\alpha - \bar{\boldsymbol{r}}_\alpha\|^2}, \tag{26}$$

where $\hat{\boldsymbol{r}}_\alpha$ is the reconstructed position of the αth point, and $\bar{\boldsymbol{r}}_\alpha$ its true position. The table in Fig. 4 lists the values for unconstrained triangulation [12], the first approximation of the planar triangulation (the iteration is terminated after the first round), which corresponds to the result of Kanazawa and Kanatani [13], and the exact values computed by our method.

From this table, we observe that the reprojection error e *increases* by assuming planarity. This is because the corresponding points need to be displaced so that the rays not simply intersect but also intersect on the specified plane. Statistical analysis [8] tells us that under maximum likelihood Ne^2/σ^2 is subject to a χ^2 distribution with N degree of freedom for unconstrained triangulation and with $2N$ degrees of freedom for planar triangulation. Hence, e should be approximately σ and $\sqrt{2}\sigma$, respectively. The values in the table in Fig. 4 are very close to the prediction. However, the increase in the reprojection error e does *not* mean the
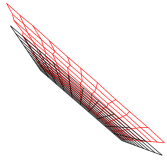
plane	reprojection error	theoretical expectation	3-D error
known	2.83124	2.82843	0.01919
unknown	1st approx. 2.81321	—	0.13643
	exact 2.81323	2.78128	0.13660

Fig. 5 The reconstructed grid by estimating the plane and the camera positions (in red) and its true position (in black). The table lists the reprojection error of planar triangulation by estimating the plane and the camera positions, its theoretical expectation, and the average 3-D reconstruction error.

increase in the 3-D reconstruction error D. In fact, the 3-D reconstruction error D actually decreases with the knowledge of planarity. We also see that the exact values are very close to the first approximation of Kanazawa and Kanatani[13].

Next, we tested the case in which the plane and camera parameters are unknown. Since the absolute scale is indeterminate, we scaled the relative displacement between the cameras to unit length. **Figure 5** shows the 3-D positions of the reconstructed grid (in red) and its true position (in black). Due to the error in estimating the plane, i.e., the homography, the computed position is slightly different from its true position. The table in Fig. 5 compares the reprojection error e and the 3-D reconstruction error D in the known and unknown plane cases. The values in the known plane case are the same as in the table in Fig. 4 except the normalization $\|\boldsymbol{t}\| = 1$.

We observe that the reprojection error e is *smaller* in the unknown plane case than in the known plane case. This is because the parameters of the plane are estimated so that the reprojection error is minimized. Statistical analysis[8] tells us that under maximum likelihood Ne^2/σ^2 is subject to a χ^2 distribution with $2N - 8$ degrees of freedom and hence has expectation $2N - 8$. This reflects the fact that the homography constraint has eight degrees of freedom with codimension two[8]. Consequently, the reprojection error e should approximately be $\sqrt{2(1 - 4/N)}\sigma$. The value in the table in Fig. 4 is very close to the prediction. We also see that the first approximation (using only a single Sampson error minimization step) and the exact maximum likelihood value are very close to each other, as generally predicted in Ref. 10). Again, the smaller reprojection error does not mean increase in 3-D reconstruction accuracy. Rather, the 3-D recon-

struction accuracy deteriorates because of the error in estimating the plane, as shown in the table in Fig. 4.

Note that when the camera positions are unknown, the 3-D positions of the points cannot be reconstructed without the knowledge of planarity. If the points are in general position, their 3-D positions and the camera positions can be reconstructed from two views[11], but that computation fails if the points degenerate to be coplanar[4],[8].

7. Concluding Remarks

We have presented an optimal algorithm[⋆1] for computing the 3-D positions of points viewed from two images using the knowledge that they are constrained to be on a planar surface. This is an extension of the unconstrained triangulation of Kanatani et al.[12] which does not assume planarity. Our algorithm automatically encompass the case in which the plane and camera parameters are unknown; they are estimated merely by inserting a Sampson error minimization step. As a result, an exact maximum likelihood estimate is obtained for the homography between the two images. Note that the homography is optimally estimated even if the points we are observing are not pecisely coplanar in the scene, so this algorithm has a wide range of applications involving not exactly planar surfaces such as building facades and road surfaces.

Our algorithm here is a complete parallel to the scheme of Kanatani and Sugaya[9] for computing an exact maximum likelihood estimate of the fundamental matrix between two images merely by inserting a Sampson error minimization step in the unconstrained triangulation of Kanatani et al.[12]. In contrast, the optimal triangulation of Hartley and Sturm[5], which solves a 6th degree polynomial, is not so easily converted to optimal fundamental matrix estimation. Similarly, the optimal planar triangulation of Chum et al.[3], which solves an 8th degree polynomial, is not so easily converted to produce an optimal homography.

We have also confirmed experimentally that the first approximation is very close to the exact maximum likelihood estimate. Thus, we may say that our optimal scheme is not really necessary in practice. In fact, the Sampson error

⋆1 The code is available at http://www.suri.cs.okayama-u.ac.jp/~kanatani/e/.

minimization solution is known to coincide with the exactly optimal solution up to several significant digits in many problems [10], and a variaty of real image applications have been published in the literature [1],[3],[4],[8],[13]. This is the reason we do not show real image demos in this paper. Rather, we stress the theoretical importance of our formulation: The performance of existing methods can only be evaluated by comparing it with our optimal solution.

In our experiment, we used the least squares, also known as the DLT, to compute the initial value h to start the FNS iterations described in Section 5. We have observed that the iterations may not converge in the presence of extremely large noise and that the use of "HyperLS" or its "Taubin approximation" [17] can significantly extend the noise level range of convergence.

Our optimal homography computation does not reach a global minimum of the reprojection error if Step (5) does not return a global minimum of the Sampson error. In fact, the FNS described in Section 5 is not theoretically guaranteed to return a global minimum of the Sampson error J, although no counterexample has been witnessed in our experiences. If the Sampson error J could be globally minimized in each iteration, e.g., using branch and bound, we would obtain a global minimum solution h in the end, because the Sampson error J coincides with the reprojection error when the iterations have converged [10].

References

1) Bartoli, A. and Sturm, P.: Constrained structure and motion from multiple uncalibrated views of a piecewise planar scene, *Int. J. Comput. Vis.*, Vol.52, No.1, pp.45–64 (2003).
2) Chojnacki, W., Brooks, M.J., van den Hengel, A. and Gawley, D.: On the fitting of surfaces to data with covariances, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.22, No.11, pp.1294–1303 (2000).
3) Chum, O., Pajdla, T. and Sturm, P.: The geometric error for homographies, *Comput. Vis. Im. Understand.*, Vol.97, No.1, pp.86–102 (2002).
4) Hartley, R. and Zisserman, A.: *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, Cambridge (2004).
5) Hartley, R.I. and Sturm, P.: Triangulation, *Comput. Vision Image Understand.*, Vol.68, No.2, pp.146–157 (1997).
6) Hay, J.C.: Optical motions and space perception—An extension of Gibson's analysis, *Psych. Rev.*, Vol.73, No.6, pp.550–565 (1966).
7) Kanatani, K.: *Geometric Computation for Machine Vision*, Oxford University Press, Oxford (1993).
8) Kanatani, K.: *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier, Amsterdam (1996); reprinted, Dover, York (2005).
9) Kanatani, K. and Sugaya, Y.: Compact fundamental matrix computation, *IPSJ Trans. Comp. Vis. Appl.*, Vol.2, pp.59–70 (2010).
10) Kanatani, K. and Sugaya, Y.: Unified computation of strict maximum likelihood for geometric fitting, *J. Math. Imaging Vis.*, Vol.38, No.1, pp.1–13 (2010).
11) Kanatani, K., Sugaya, Y. and Kanazawa, Y.: Latest algorithms for 3-D reconstruction form two views. Chen, C.H. (Ed.), *Handbook of Pattern Recognition and Computer Vision*, 4th ed., World Scientific, Singapore, pp.201–234 (2009).
12) Kanatani, K., Sugaya, Y. and Niitsuma, H.: Triangulation from two views revisited: Hartley-Sturm vs. optimal correction, *Proc. 9th British Machine Vis. Conf.*, Leeds, U.K., September 2008, pp.173–182 (2008).
13) Kanazawa, Y. and Kanatani, K.: Direct reconstruction of planar surfaces by stereo vision, *IEICE Trans. Inf. Syst.*, Vol.E78-D, No.7, pp.917–922 (1995).
14) Kanazawa, Y. and Kanatani, K.: Reliability of 3-D reconstruction by stereo vision, *IEICE Trans. Inf. Syst.*, Vol.E78-D, No.10, pp.1301–1306 (1995).
15) Lindstrom, P.: Triangulation made easy, *Proc. IEEE Conf. Comput. Vis. Patt. Recog.*, San Fransico, CA, U.S.A. June 2010 (2010).
16) Longuet-Higgins, H.C.: The reconstruction of a planar surface from two perspective projections, *Proc. Royal Soc. Lond.*, Vol.B-227 (1249), pp.399–410 (1986).
17) Niitsuma, H., Rangarajan, P. and Kanatani, K.: High accuracy homography computation without iterations, *Proc. 16th Symp. Sensing via Imaging Information*, Yokohama, Japan, June 2010 (2010).
18) Scoleri, T., Chojnacki, W. and Brooks, M.J.: A multi-objective parameter estimator for image mosaicing, *Proc. 8th Int. Symp. Signal Processing and its Applications*, Sydney, Australia, August 2005, Vol.2, pp.551–554 (2005).
19) Tossavainen, T.: Approximate and SQP two view triangulation, *Proc. 10th Asian Conf. Comput. Vis.*, Queenstown, New Zealand, November 2010, Vol.3, pp.1303–1316 (2010).
20) Tsai, R.Y. and Huang, T.S.: Estimation of three-dimensional motion parameters of a rigid planar patch, *IEEE Trans. Acoustics Speech Signal Process.*, Vol.29, No.6, pp.1147–1152 (1981).
21) Tsai, R.Y. and Huang, T.S.: Estimation of three-dimensional motion parameters of a rigid planar patch III: Finite point correspondence and the three-view problem, *IEEE Trans. Acoustics Speech Signal Process.*, Vol.32, No.2, pp.213–220 (1984).
22) Tsai, R.Y., Huang, T.S. and Zhu, W.-L.: Estimation of three-dimensional motion

parameters of a rigid planar patch II: Singular value decomposition, *IEEE Trans. Acoustics Speech Signal Process.*, Vol.30, No.4, pp.525–544 (1982).

Appendix

A.1 Derivation of optimal planar triangulation

Instead of computing $\bar{\boldsymbol{p}}$ that minimizes Eq. (7), we can let $\bar{\boldsymbol{p}} = \boldsymbol{p} - \Delta\boldsymbol{p}$ and compute $\Delta\boldsymbol{p}$. Equations (7) and (8) are now rewritten as

$$E = \|\Delta\boldsymbol{p}\|^2, \qquad (\boldsymbol{\xi}^{(k)}(\boldsymbol{p} - \Delta\boldsymbol{p}), \boldsymbol{h}) = 0, \qquad k = 1, 2, 3. \tag{27}$$

Writing $\boldsymbol{\xi}^{(k)} = \boldsymbol{\xi}^{(k)}(\boldsymbol{p})$, substituting the Taylor expansion $\boldsymbol{\xi}^{(k)}(\boldsymbol{p} - \Delta\boldsymbol{p}) = \boldsymbol{\xi}^{(k)} - \boldsymbol{T}^{(k)}\Delta\boldsymbol{p} + \cdots$ to the second of Eqs. (27), and omitting second order terms in $\Delta\boldsymbol{p}$, we obtain to a first approximation

$$(\boldsymbol{T}^{(k)}\Delta\boldsymbol{p}, \boldsymbol{h}) = (\boldsymbol{\xi}^{(k)}, \boldsymbol{h}). \tag{28}$$

Introducing Lagrange multipliers $\lambda^{(k)}$, differentiating

$$\frac{1}{2}\|\Delta\boldsymbol{p}\|^2 - \sum_{k=1}^{3} \lambda^{(k)}(\boldsymbol{T}^{(k)}\Delta\boldsymbol{p}, \boldsymbol{h}) = \frac{1}{2}(\Delta\boldsymbol{p}, \Delta\boldsymbol{p}) - \sum_{k=1}^{3} \lambda^{(k)}(\Delta\boldsymbol{p}, \boldsymbol{T}^{(k)\top}\boldsymbol{h}) \tag{29}$$

with respect to $\Delta\boldsymbol{p}$, setting the result to $\boldsymbol{0}$, we obtain

$$\Delta\boldsymbol{p} = \sum_{k=1}^{3} \lambda^{(k)} \boldsymbol{T}^{(k)\top}\boldsymbol{h}. \tag{30}$$

Hence, $E = \|\Delta\boldsymbol{p}\|^2$ has the form

$$E = \left(\sum_{k=1}^{3} \lambda^{(k)} \boldsymbol{T}^{(k)\top}\boldsymbol{h}, \sum_{l=1}^{3} \lambda^{(l)} \boldsymbol{T}^{(l)\top}\boldsymbol{h} \right) = \sum_{k,l=1}^{3} V^{(kl)}\lambda^{(k)}\lambda^{(l)}, \tag{31}$$

where $V^{(kl)} = (\boldsymbol{h}, V_0^{(kl)}[\boldsymbol{\xi}]\boldsymbol{h})$ (recall the definition of $V_0^{(kl)}[\boldsymbol{\xi}]$ in Eq. (14)). Substituting Eq. (30) into Eq. (28), we obtain

$$\sum_{l=1}^{3} (\boldsymbol{h}, V_0^{(kl)}[\boldsymbol{\xi}]\boldsymbol{h})\lambda^{(l)} = (\boldsymbol{\xi}^{(k)}, \boldsymbol{h}). \tag{32}$$

This defines a set of simultaneous linear equations in $\lambda^{(l)}$, but the coefficient matrix has rank 2 in the absence of noise. So, we solve it by using the pseudoinverse

$\boldsymbol{W}$ of truncated rank 2 in Eq. (15) and obtain

$$\lambda^{(k)} = \sum_{l=1}^{3} W^{(kl)}(\boldsymbol{\xi}^{(l)}, \boldsymbol{h}). \tag{33}$$

Substitution of this into Eq. (31) yields

$$\begin{aligned} E &= \sum_{k,l=1}^{3} V^{(kl)} \sum_{m=1}^{3} W^{(km)}(\boldsymbol{\xi}^{(m)}, \boldsymbol{h}) \sum_{n=1}^{3} W^{(ln)}(\boldsymbol{\xi}^{(n)}, \boldsymbol{h}) \\ &= \sum_{k,l=1}^{3} W^{(kl)}(\boldsymbol{\xi}^{(k)}, \boldsymbol{h})(\boldsymbol{\xi}^{(l)}, \boldsymbol{h}), \end{aligned} \tag{34}$$

where we have used the identity $\boldsymbol{W}\boldsymbol{V}\boldsymbol{W} = \boldsymbol{W}(\boldsymbol{W})_2^{-}\boldsymbol{W} = \boldsymbol{W}$ for pseudoinverses. The true value $\bar{\boldsymbol{p}}$ is now estimated from Eqs. (30) and (33) by replacing $\boldsymbol{\xi}^{(l)*}$ by $\boldsymbol{\xi}^{(l)}$ in Eq. (16).

This is only a first approximation. So, we let $\bar{\boldsymbol{p}} = \hat{\boldsymbol{p}} - \Delta\hat{\boldsymbol{p}}$ and estimate the higher order correction $\Delta\hat{\boldsymbol{p}}$. Now, Eqs. (7) and (8) are rewritten as

$$E = \|\tilde{\boldsymbol{p}} + \Delta\hat{\boldsymbol{p}}\|^2, \qquad (\boldsymbol{\xi}^{(k)}(\hat{\boldsymbol{p}} - \Delta\hat{\boldsymbol{p}}), \boldsymbol{h}) = 0, \qquad k = 1, 2, 3, \tag{35}$$

where we put $\tilde{\boldsymbol{p}} = \boldsymbol{p} - \hat{\boldsymbol{p}}$. Substituting the Taylor expansion of $\boldsymbol{\xi}^{(k)}(\hat{\boldsymbol{p}} - \Delta\hat{\boldsymbol{p}})$ and omitting second order terms in the higher order correction $\Delta\hat{\boldsymbol{p}}$, we obtain

$$(\hat{\boldsymbol{T}}^{(k)}\Delta\hat{\boldsymbol{p}}, \boldsymbol{h}) = (\hat{\boldsymbol{\xi}}^{(k)}, \boldsymbol{h}), \tag{36}$$

where $\hat{\boldsymbol{\xi}}^{(k)} = \boldsymbol{\xi}^{(k)}(\hat{\boldsymbol{p}})$ and $\hat{\boldsymbol{T}}^{(k)}$ is the value of the matrix $\boldsymbol{T}^{(k)}$ obtained by substituting $\boldsymbol{p} = \hat{\boldsymbol{p}}$. Introducing Lagrange multipliers λ, differentiating

$$\begin{aligned} \frac{1}{2}\|\tilde{\boldsymbol{p}} + \Delta\hat{\boldsymbol{p}}\|^2 - \sum_{k=1}^{3} \lambda^{(k)}(\hat{\boldsymbol{T}}^{(k)}\Delta\hat{\boldsymbol{p}}, \boldsymbol{h}) &= \frac{1}{2}(\tilde{\boldsymbol{p}} + \Delta\hat{\boldsymbol{p}}, \tilde{\boldsymbol{p}} + \Delta\hat{\boldsymbol{p}}) \\ &\quad - \sum_{k=1}^{3} \lambda^{(k)}(\Delta\hat{\boldsymbol{p}}, \hat{\boldsymbol{T}}^{(k)\top}\boldsymbol{h}) \end{aligned} \tag{37}$$

with respect to $\Delta\hat{\boldsymbol{p}}$, and setting the result to $\boldsymbol{0}$, we obtain

$$\Delta\hat{\boldsymbol{p}} = \sum_{k=1}^{3} \lambda^{(k)} \hat{\boldsymbol{T}}^{(k)\top}\boldsymbol{h} - \tilde{\boldsymbol{p}}. \tag{38}$$

Substituting of this into $E = \|\tilde{\boldsymbol{p}} + \Delta\hat{\boldsymbol{p}}\|^2$, we have

$$E = \left(\sum_{k=1}^{3} \lambda^{(k)} \hat{\boldsymbol{T}}^{(k)\top} \boldsymbol{h}, \sum_{l=1}^{3} \lambda^{(l)} \hat{\boldsymbol{T}}^{(l)\top} \boldsymbol{h}\right) = \sum_{k,l=1}^{3} \hat{V}^{(kl)} \lambda^{(k)} \lambda^{(l)}, \tag{39}$$

where we put $\hat{V}_0^{(kl)}[\boldsymbol{\xi}] = \hat{\boldsymbol{T}}^{(k)} \hat{\boldsymbol{T}}^{(l)\top}$ and $\hat{V}^{(kl)} = (\boldsymbol{h}, \hat{V}_0^{(kl)}[\boldsymbol{\xi}]\boldsymbol{h})$. Substituting Eq. (38) into Eq. (36), we obtain after rearrangement

$$\sum_{l=1}^{3} \hat{V}^{(kl)} \lambda^{(l)} = (\hat{\boldsymbol{\xi}}^{(k)*}, \boldsymbol{h}), \tag{40}$$

where we put $\hat{\boldsymbol{\xi}}^{(k)*} = \hat{\boldsymbol{\xi}}^{(k)} + \hat{\boldsymbol{T}}^{(k)} \tilde{\boldsymbol{p}}$. From Eq. (40), the Lagrange multipliers $\lambda^{(k)}$ are obtained in the form

$$\lambda^{(k)} = \sum_{l=1}^{3} \hat{W}^{(kl)} (\hat{\boldsymbol{\xi}}^{(l)*}, \boldsymbol{h}), \tag{41}$$

where $\hat{W}^{(kl)}$ is the (kl) element of the pseudoinverse $\hat{\boldsymbol{W}} = (\hat{\boldsymbol{V}})_2^-$ of truncated rank 2 of the matrix $\hat{\boldsymbol{V}}$. Substituting Eq. (41) into Eq. (39), we can write the reprojection error E in the form

$$E = \sum_{k,l=1}^{3} \hat{V}^{(kl)} \sum_{m=1}^{3} \hat{W}^{(km)} (\hat{\boldsymbol{\xi}}^{(m)*}, \boldsymbol{h}) \sum_{n=1}^{3} \hat{W}^{(ln)} (\hat{\boldsymbol{\xi}}^{(n)*}, \boldsymbol{h})$$

$$= \sum_{k,l=1}^{3} \hat{W}^{(kl)} (\hat{\boldsymbol{\xi}}^{(k)*}, \boldsymbol{h})(\hat{\boldsymbol{\xi}}^{(l)*}, \boldsymbol{h}), \tag{42}$$

where we have used the identity $\hat{\boldsymbol{W}}\hat{\boldsymbol{V}}\hat{\boldsymbol{W}} = \hat{\boldsymbol{W}}(\hat{\boldsymbol{W}})_2^- \hat{\boldsymbol{W}} = \hat{\boldsymbol{W}}$ of pseudoinverses. From Eqs. (38) and (39), the true value $\bar{\boldsymbol{p}}$ can be estimated by Eq. (16). We let the result be $\hat{\boldsymbol{p}}$ and estimate the yet higher correction term by the same procedure. In the end, $\Delta\hat{\boldsymbol{p}}$ converges to $\boldsymbol{0}$.

A.2 Plane and motion parameters from homography

Since the XYZ world coordinate system can be arbitrarily defined in the scene, we identify it with the first camera coordinate system so that the motion parameters of the first camera are $\{\boldsymbol{I}, \boldsymbol{0}\}$; let $\{\boldsymbol{R}, \boldsymbol{t}\}$ be the parameters of the second. The absolute scale of the scene cannot be determined from images, so we let $\|\boldsymbol{t}\| = 1$. The procedure for computing the parameters $\{\boldsymbol{n}, d\}$ and $\{\boldsymbol{R}, \boldsymbol{t}\}$ from a given homography $\boldsymbol{H}$ has be been presented by many researchers in many different forms [6),16),20)–22)]. The following is a modification of the formulation in Ref. 7):

(1) Let $\tilde{\boldsymbol{H}}$ be

$$\tilde{\boldsymbol{H}} = \mathrm{diag}\left(1, 1, \frac{f_2}{f_0}\right) \boldsymbol{H} \mathrm{diag}\left(1, 1, \frac{f_0}{f_1}\right). \tag{43}$$

(2) Normalize $\tilde{\boldsymbol{H}}$ to have determinant 1:

$$\tilde{\boldsymbol{H}} \leftarrow \frac{\tilde{\boldsymbol{H}}}{\sqrt[3]{\det[\tilde{\boldsymbol{H}}]}}. \tag{44}$$

(3) Compute the singular value decomposition (SDV) of $\tilde{\boldsymbol{H}}$ in the form ($\boldsymbol{V}$ and $\boldsymbol{V}$ are orthogonal matrices):

$$\tilde{\boldsymbol{H}} = \boldsymbol{V} \mathrm{diag}(\sigma_1, \sigma_2, \sigma_3) \boldsymbol{V}^{\top}. \tag{45}$$

(4) Let $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$ be the columns of the matrix $\boldsymbol{V}$. The paremeters $\{\boldsymbol{n}, d\}$ are given by

$$\boldsymbol{n} = \mathcal{N}\left[\sqrt{\sigma_1^2 - \sigma_2^2}\boldsymbol{v}_1 \pm \sqrt{\sigma_2^2 - \sigma_3^2}\boldsymbol{v}_3\right], \qquad d = \frac{\sigma_2}{\sigma_1 - \sigma_3}, \tag{46}$$

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

(5) Compute $\boldsymbol{t}$ and $\boldsymbol{R}$ by

$$\boldsymbol{t} = \mathcal{N}\left[-\sigma_3\sqrt{\sigma_1^2 - \sigma_2^2}\boldsymbol{v}_1 \pm \sigma_1\sqrt{\sigma_2^2 - \sigma_3^2}\boldsymbol{v}_3\right], \quad \boldsymbol{R} = \frac{1}{\sigma_2}\left(\boldsymbol{I} + \frac{\sigma_2^3 \boldsymbol{n}\boldsymbol{t}^{\top}}{d}\right)\tilde{\boldsymbol{H}}^{\top}, \tag{47}$$

where the double sign $\pm$ has the same order as in Eq. (46).

(6) Another solution is obtained by simultaneously changing the sign of both $\boldsymbol{n}$ and $\boldsymbol{t}$.

Thus, we obtain four sets of solutions for $\boldsymbol{t}$, $\boldsymbol{R}$, d and $\boldsymbol{n}$, but they include geometrically identical solutions, so we obtain two sets of geometrically different solutions. It is easy to pick out a correct solution, since in realistic situations the incorrect solution has a very unnatural geometry. As an easy criterion, consider for example the vanishing line of the plane, which may not exist within the image framework. It divides the image plane into a plane region and an nonplane region (e.g., the background). If the image origin is within the plane region, which is

usually the case, we can select the correct solution by choosing the one for which the Z components of vectors $\boldsymbol{R}_1^\top \boldsymbol{n}/d$ and $\boldsymbol{R}_2^\top \boldsymbol{n}/d$ are both positive. If there is no such solution, we reconstruct the 3-D positions of the corresponding points using our planar triangulation algorithm for both solutions and choose the one for which the computed 3-D positions are in front of both cameras [7].

A.3 Sampson error and optimal homography estimation

The procedure in Appendix A.1 orthogonally projects the 4-D joint point $\boldsymbol{p} = (x, y, x', y')^\top$ onto the 2-D variaty $\mathcal{S}$ in the 4-D space. First, the projection direction is determined from the coordinates (x, y, x', y') of $\boldsymbol{p}$ as a first approximation to the correct orthogonal direction. With this projection, the reprojection error is given by Eq. (34). If we do this for all the N points $\boldsymbol{p}_\alpha = (x_\alpha, y_\alpha, x'_\alpha, y'_\alpha)^\top$, $\alpha = 1, \ldots, N$, the sum of the reprojection errors divided by N is

$$J = \frac{1}{N} \sum_{\alpha=1}^{N} \sum_{k,l=1}^{3} W_\alpha^{(kl)} (\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{h})(\boldsymbol{\xi}_\alpha^{(l)}, \boldsymbol{h}), \tag{48}$$

which is known as the *Sampson error* [4]. So far, we have assumed that the homography $\boldsymbol{h}$ is given. If $\boldsymbol{h}$ is not given, it is reasonable to choose such $\boldsymbol{h}$ that minimizes Eq. (48). Now, suppose the homography $\boldsymbol{h}$ is so chosen from the beginning. Since the first projection direction may not be exactly orthogonal to $\mathcal{S}$ we correct it as described in Appendix A.1. As a result, the reprojection error for each $\boldsymbol{p}$ is given by Eq. (42). Summing it for all α and dividing it by N, we obtain

$$J = \frac{1}{N} \sum_{\alpha=1}^{N} \sum_{k,l=1}^{3} W_\alpha^{(kl)} (\boldsymbol{\xi}_\alpha^{(k)*}, \boldsymbol{h})(\boldsymbol{\xi}_\alpha^{(l)*}, \boldsymbol{h}). \tag{49}$$

Then, we update the homograpy $\boldsymbol{h}$ by choosing the value that minimized this expression. Note that Eq. (49) has the same form as Eq. (48) except that $\boldsymbol{\xi}_\alpha^{(k)}$ in Eq. (48) are replaced by $\boldsymbol{\xi}_\alpha^{(k)*}$. Hence, if a minimization tool for Eq. (48) is available, it can also be used to minimize Eq. (49).

After the iterations have converged, the projection of each $\boldsymbol{p}_\alpha$ is orthogonal to the variety $\mathcal{S}$ detemined by the homography $\boldsymbol{h}$ that minimizes the sum of the reprojection errors for all the corresponding pairs of points. Thus, we obtain the procedure described in Section 5.

A.4 Sampson error minimization for homography

As pointed out above, it is sufficient to derive a scheme for minimizing Eq. (48); Eq. (49) is minimized using the same scheme.

Since the identity $x'\boldsymbol{\xi}^{(1)} + y'\boldsymbol{\xi}^{(2)} + f_0\boldsymbol{\xi}^{(3)} = \boldsymbol{0}$ holds for $\boldsymbol{\xi}^{(k)}$ in Eqs. (7) as mentioned in Section 3, we obtain

$$(x'_\alpha \boldsymbol{\xi}_\alpha^{(1)} + y'_\alpha \boldsymbol{\xi}_\alpha^{(2)} + f_0 \boldsymbol{\xi}_\alpha^{(3)}, \boldsymbol{h}) = 0, \tag{50}$$

for each α. Since this is an identity in x_α, y_α, x'_α, and y'_α, its derivatives with respect to these are also identities. Hence, the following identically hold if there is no noise:

$$\begin{aligned}
(x'_\alpha [\boldsymbol{T}_\alpha^{(1)}]_1 + y'_\alpha [\boldsymbol{T}_\alpha^{(2)}]_1 + f_0 [\boldsymbol{T}_\alpha^{(3)}]_1, \boldsymbol{h}) = 0, \\
(x'_\alpha [\boldsymbol{T}_\alpha^{(1)}]_2 + y'_\alpha [\boldsymbol{T}_\alpha^{(2)}]_2 + f_0 [\boldsymbol{T}_\alpha^{(3)}]_2, \boldsymbol{h}) = 0, \\
(x'_\alpha [\boldsymbol{T}_\alpha^{(1)}]_3 + y'_\alpha [\boldsymbol{T}_\alpha^{(2)}]_3 + f_0 [\boldsymbol{T}_\alpha^{(3)}]_3, \boldsymbol{h}) = 0, \\
(x'_\alpha [\boldsymbol{T}_\alpha^{(1)}]_4 + y'_\alpha [\boldsymbol{T}_\alpha^{(2)}]_4 + f_0 [\boldsymbol{T}_\alpha^{(3)}]_4, \boldsymbol{h}) = 0.
\end{aligned} \tag{51}$$

Here, $[\boldsymbol{T}_\alpha^{(k)}]_i$ is the ith column of $\boldsymbol{T}_\alpha^{(k)}$, and we have noted that $(\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{h}) = 0$ in the absence of noise. From these four equations, we conclude that

$$(x'_\alpha \boldsymbol{T}_\alpha^{(1)} + y'_\alpha \boldsymbol{T}_\alpha^{(2)} + f_0 \boldsymbol{T}_\alpha^{(3)})^\top \boldsymbol{h} = \boldsymbol{0}. \tag{52}$$

If we multiply $\boldsymbol{T}_\alpha^{(k)}$ with this and note the definition $V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \equiv \boldsymbol{T}_\alpha^{(k)} \boldsymbol{T}_\alpha^{(l)\top}$, we obtain

$$(x'_\alpha V_0^{(k1)}[\boldsymbol{\xi}_\alpha] + y'_\alpha V_0^{(k2)}[\boldsymbol{\xi}_\alpha] + f_0 V_0^{(k3)}[\boldsymbol{\xi}_\alpha])\boldsymbol{h} = \boldsymbol{0}. \tag{53}$$

Let us write the 3×3 matrix having $(\boldsymbol{h}, V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\boldsymbol{h})$ as its (kl) element as $\boldsymbol{V}_\alpha$. Computing the inner product of $\boldsymbol{h}$ and Eq. (53), we obtain

$$\boldsymbol{V}_\alpha \begin{pmatrix} x'_\alpha \\ y'_\alpha \\ f_0 \end{pmatrix} = \boldsymbol{0}. \tag{54}$$

Thus, $\boldsymbol{v}'_\alpha = (x'_\alpha, y'_\alpha, f_0)^\top$ is a null vector of $\boldsymbol{V}_\alpha$. From the definition of pseudoinverse, it is also a null vector of $\boldsymbol{W}_\alpha = (\boldsymbol{V}_\alpha)_2^-$. It follows that $\boldsymbol{W}_\alpha \boldsymbol{V}_\alpha$ and $\boldsymbol{V}_\alpha \boldsymbol{W}_\alpha$ are both projection matrices onto the subspace orthogonal to $\boldsymbol{v}'_\alpha$. Hence, we can write

$$\boldsymbol{W}_\alpha \boldsymbol{V}_\alpha = \boldsymbol{V}_\alpha \boldsymbol{W}_\alpha = \boldsymbol{I} - \mathcal{N}[\boldsymbol{v}'_\alpha]\mathcal{N}[\boldsymbol{v}'_\alpha]^\top. \tag{55}$$

(Recall that $\mathcal{N}[\,\cdot\,]$ denotes normalization into unit norm.) Differentiating Eq. (55) with respect to h_i, we obtain

$$\frac{\partial \boldsymbol{V}_\alpha}{\partial h_i}\boldsymbol{W}_\alpha + \boldsymbol{V}_\alpha \frac{\partial \boldsymbol{W}_\alpha}{\partial h_i} = \boldsymbol{O}. \tag{56}$$

Multiplying this by $\boldsymbol{W}_\alpha$ from left and noting that $\partial \boldsymbol{W}_\alpha / \partial h_i$ also has $\boldsymbol{v}'_\alpha$ as its null vector and hence is invariant to the projection $\boldsymbol{W}_\alpha \boldsymbol{V}_\alpha$, we obtain the following identity:

$$\frac{\partial \boldsymbol{W}_\alpha}{\partial h_i} = -\boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i}\boldsymbol{W}_\alpha. \tag{57}$$

Now, if we define the 9×3 matrix

$$\boldsymbol{\Xi}_\alpha = \left(\begin{array}{ccc} \boldsymbol{\xi}^{(1)}_\alpha & \boldsymbol{\xi}^{(2)}_\alpha & \boldsymbol{\xi}^{(3)}_\alpha \end{array} \right), \tag{58}$$

Eq. (48) can be rewritten as follows:

$$J = \frac{1}{N}\sum_{\alpha=1}^{N}(\boldsymbol{h}, \boldsymbol{\Xi}_\alpha \boldsymbol{W}_\alpha \boldsymbol{\Xi}^\top_\alpha \boldsymbol{h}). \tag{59}$$

Differentiating this with respect to h_i and using Eq. (57), we obtain

$$\frac{\partial J}{\partial h_i} = \frac{2}{N}\sum_{\alpha=1}^{N}(\boldsymbol{\Xi}_\alpha \boldsymbol{W}_\alpha \boldsymbol{\Xi}^\top_\alpha \boldsymbol{h})_i - \frac{2}{N}\sum_{\alpha=1}^{N}\left(\boldsymbol{h}, \boldsymbol{\Xi}_\alpha \boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i}\boldsymbol{W}_\alpha \boldsymbol{\Xi}^\top_\alpha \boldsymbol{h}\right), \tag{60}$$

where $(\,\cdot\,)_i$ denotes the ith component. If we put

$$v^{(k)}_\alpha = \sum_{l=1}^{3}W^{(kl)}_\alpha(\boldsymbol{\xi}^{(l)}_\alpha, \boldsymbol{h}), \tag{61}$$

and define $\boldsymbol{v}_\alpha$ to be the 3-D vector with components $v^{(k)}_\alpha$, $k = 1, 2, 3$, Eq. (61) is written as

$$\boldsymbol{v}_\alpha = \boldsymbol{W}_\alpha \boldsymbol{\Xi}^\top_\alpha \boldsymbol{h}. \tag{62}$$

From the definition of the matrix $\boldsymbol{V}_\alpha$, we see that $\partial \boldsymbol{V}_\alpha / \partial h_i$ is a 3×3 matrix whose (kl) element is $2\sum_{j=1}^{9}V^{(kl)}_0[\boldsymbol{\xi}_\alpha]_{ij}h_j$. Hence, the last term of the right-hand side of Eq. (60) is

$$\frac{2}{N}\sum_{\alpha=1}^{N}\left(\boldsymbol{h}, \boldsymbol{\Xi}_\alpha \boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i}\boldsymbol{W}_\alpha \boldsymbol{\Xi}^\top_\alpha \boldsymbol{h}\right) = \frac{2}{N}\sum_{\alpha=1}^{N}\left(\boldsymbol{v}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i}\boldsymbol{v}_\alpha\right)$$
$$= \sum_{j=1}^{9}\Big(\frac{2}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}V^{(kl)}_0[\boldsymbol{\xi}_\alpha]_{ij}v^{(k)}_\alpha v^{(l)}_\alpha\Big)h_j. \tag{63}$$

If we define 9×9 matrices $\boldsymbol{M}$ and $\boldsymbol{L}$ by

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N}W^{(kl)}_\alpha \boldsymbol{\xi}^{(k)}_\alpha \boldsymbol{\xi}^{(l)\top}_\alpha, \qquad \boldsymbol{L} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{N}v^{(k)}_\alpha v^{(l)}_\alpha V^{(kl)}_0[\boldsymbol{\xi}_\alpha], \tag{64}$$

the first term on the right-hand side of Eq. (60) is simply $2\boldsymbol{M}$. Equation (63) is written as $2\boldsymbol{L}\boldsymbol{h}$. Thus, we obtain the following expression of the derivative of J in Eq. (59):

$$\nabla_{\boldsymbol{h}}J = 2(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{h}. \tag{65}$$

It follows that to minimize J we need to solve

$$(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{h} = \boldsymbol{0}. \tag{66}$$

In the above derivation, we have assumed that there is no noise. In the presence of noise, the only difference is that Eq. (54) does not exactly hold, and $\boldsymbol{V}_\alpha$ is nonsingular with the smallest eigenvalue close to 0. So, we regard the definition of $\boldsymbol{W}_\alpha = (\boldsymbol{V}_\alpha)^-_2$ as obtained by truncating the smallest eigenvalue of $\boldsymbol{V}_\alpha$ to 0. Using the FNS principle of Chojnacki et al. [2], we obtain the following procedure:

(1) Provide an initial value $\boldsymbol{h}_0$ for $\boldsymbol{h}$.

(2) Compute the matrices $\boldsymbol{M}$ and $\boldsymbol{L}$ in Eqs. (64).

(3) Solve the eigenvalue problem

$$(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{h} = \lambda \boldsymbol{h}, \tag{67}$$

and compute the unit eigenvector $\boldsymbol{h}$ for the smallest eigenvalue λ.

(4) If $\boldsymbol{h} \approx \boldsymbol{h}_0$, return $\boldsymbol{h}$ and stop. Else, let $\boldsymbol{h}_0 \leftarrow \boldsymbol{h}$, and go back to Step (2).

Recall that the (kl) elements of $\boldsymbol{V}_\alpha$ is $(\boldsymbol{h}, V^{(kl)}_0[\boldsymbol{\xi}_\alpha]\boldsymbol{h})$, so if we multiply $\boldsymbol{h}$ by a constant c, the matrix $\boldsymbol{V}_\alpha$ is multiplied by c^2. Hence, the matrix $\boldsymbol{W}_\alpha = (\boldsymbol{V}_\alpha)^-_2$ is multiplied by $1/c^2$. This means that J in Eq. (59) is invariant to the scale change of $\boldsymbol{h}$. Hence, its gradient $\nabla_{\boldsymbol{h}}J$ is orthogonal to $\boldsymbol{h}$, so $(\boldsymbol{h}, (\boldsymbol{M} - \boldsymbol{L})\boldsymbol{h}) = 0$ for any $\boldsymbol{h}$. It follows that when the FNS iterations have converged to some $\boldsymbol{h}$ that satisfies Eq. (67), λ is necessarily 0 and Eq. (66) holds.

In the extremely noisy case, the FNS iterations may not converge. In such a case, replacing $\boldsymbol{h} \leftarrow \boldsymbol{h}_0$ by $\boldsymbol{h} \leftarrow \mathcal{N}[\boldsymbol{h} + \boldsymbol{h}_0]$ $(= \mathcal{N}[(\boldsymbol{h} + \boldsymbol{h}_0)/2])$ is effective for accelerating the convergence [9]. The use of "HyperLS" or its "Taubin approximation" [17] for computing the initial value is also effective, as pointed out in Section 7.

(Communicated by *Long Quan*)

Kenichi Kanatani received his B.E., M.S., and Ph.D. in applied mathematics from the University of Tokyo in 1972, 1974 and 1979, respectively. After serving as Professor of computer science at Gunma University, Gunma, Japan, he is currently Professor of computer science at Okayama University, Okayama, Japan. He is the author of many books on computer vision and received many awards including the Best Paper Awards from IPSJ (1987) and IEICE (2005). He is an IEEE Fellow.

Hirotaka Niitsuma received his B.E. an M.S. from Osaka University, Japan, in 1993 and 1995, respectively, and his Ph.D. from NAIST, Japan, in 1999. He was a researcher at TOSHIBA, at JST Corporation, at Denso IT Laboratory, Inc., at Kwansei Gakuin University, Japan, at Kyungpook National University, Korea, and at AIST, Japan. From April 2007, he is Assistant Professor of computer science at Okayama University, Japan. His research interests include computer vision, machine learning, and neural networks.