High Accuracy Homography Computation without Iterations

Hirotaka Niitsuma†, Prasanna Rangarajan‡, and Kenichi Kanatani†
†Okayama University, Japan, ‡Southern Methodist University, U.S.A.
E-mail: {niitsuma,kanatani}@suri.cs.okayama-u.ac.jp,prangara@mail.sum.ed

Abstract. We present a highly accurate least-squares (LS) alternative to the theoretically optimal maximum likelihood (ML) estimator for homographies between two images. Unlike ML, our estimator is non-iterative and yields a solution even in the presence of large noise. By rigorous error analysis, we derive a "hyperLS" estimator which is unbiased up to second order noise terms. We also introduce a computational simplification, which we call "Taubin approximation", without incurring an accuracy loss. We experimentally demonstrate that our estimator far surpasses the standard LS and is nearly comparable to the ML and the theoretical accuracy limit (the KCR lower bound).

1. Introduction

Computing a homography between two images (Fig. 1) is the first step in many computer vision applications including panoramic image generation, camera calibration using reference planes, 3-D reconstruction of objects that have planar faces, and detecting obstacles on a planar surface.

The simplest and most widely used method for estimating homographies is the least squares (LS), which minimizes the sum of squares of the constraint equations, known as the "algebraic distance" [3]. However, its accuracy is limited in the presence of noise. A more accurate solution can be obtained by maximum likelihood (ML), which under independent and isotropic Gaussian noise reduces to minimizing of the "reprojection error", or the "geometric distance", subject to the homography constraint. This is also known as the "Gold standard" [3]. However, all ML-based estimators are iterative and may not converge for very large noise. In addition, an appropriate initial guess is needed to start the iterations. Thus, an accurate algebraic estimator which yields analytical solutions is desired, even if it is not strictly optimal.

Similar circumstances arise in other problems including fitting a circle/ellipse to a noisy point sequence and estimating fundamental matrices from noisy point correspondences. For these problems, the Taubin estimator [16] has emerged as an algebraic alternative with accuracy comparable to ML [6, 9]. However, the Taubin estimator is defined only for a single constraint, such as the circle/ellipse equation and the epipolar equation, while a homography is described by multiple equations. It was only recently that Rangarajan and Papamichalis [14] revealed the existence of a "Taubin-like" estimator for homographies, but they failed to rigorously analyze the accuracy of their estimator.

On the other hand, Al-Sharadqah and Chernov [1], Rangarajan and Kanatani [13], and Kanatani and Rangarajan [8] recently proposed a very accurate LS estimator for circle and ellipse fitting based on the perturbation theory of Kanatani [6]; it eliminates the bias of the fitted circle/ellipse up to second order noise terms. In this paper, we extend their technique to homographies. The major difference between circle/ellipse fitting and homography estimation is, as we show later, that a circle is represented by a quadratic polynomial, while a homography is represented by a set of bilinear polynomials. Consequently, the bias due to the nonlinearity of the constraint is smaller for homographies than for circles and ellipses.

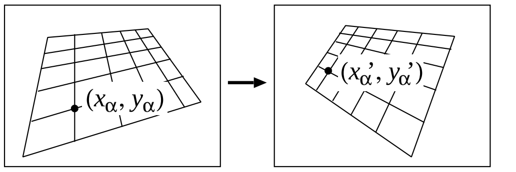

Figure 1 Computing a homography between two images.

The purpose of this paper is not to introduce a new method with higher accuracy than existing iterative ML-based methods. Rather, we present a *best* method *within the framework of algebraic distance minimization* that does not require iterations. By numerical experiments, we show that our method, which we call *hyperLS*, has accuracy nearly as high as, if not higher than, ML-based methods. In practice, our solution is best suited to initialize iterations of ML-based methods, greatly improving the convergence properties.

We summarize mathematical fundamentals in Sec. 2 and describe the principle of algebraic distance minimization in Sec. 3. In Sec. 4 and 5, we do rigorous error analysis of algebraic distance minimization by invoking the perturbation theory of Kanatani [6]. We evaluate the covariance and the bias of the solution in Sec. 6 and derive our hyperLS in Sec. 7. In Sec. 8, we do numerical simulation and demonstrate that our hyperLS is far more accurate than the standard LS. We also show the accuracy of our estimator is close to that of ML-based methods and the theoretical accuracy limit called the KCR lower bound [4, 5, 6].

2. Homography

A *homography* is an image mapping in the form

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

where f_0 is a scale constant chosen so that all terms have nearly an equal magnitude; its absence would incur serious accuracy loss in finite precision numerical computation. If we define 3-D homogeneous coordinate vectors

$$\boldsymbol{x} = \begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix}, \quad \boldsymbol{x}' = \begin{pmatrix} x'/f_0 \\ y'/f_0 \\ 1 \end{pmatrix}, \tag{2}$$

Eqs. (1) can be equivalently written as

$$\boldsymbol{x}' \cong \boldsymbol{H}\boldsymbol{x}, \tag{3}$$

where $\boldsymbol{H}$ is a 3×3 matrix with elements h_{ij}, and $\cong$ denotes equality up to a nonzero constant. Equation (3) states that vectors $\boldsymbol{x}'$ and $\boldsymbol{H}\boldsymbol{x}$ are parallel, so we can equivalently write this as

$$\boldsymbol{x}' \times \boldsymbol{H}\boldsymbol{x} = \boldsymbol{0}. \tag{4}$$

If we define 9-D vectors

$$\begin{aligned} \boldsymbol{\xi}^{(1)} &= \begin{pmatrix} 0 & 0 & 0 & -f_0 x & -f_0 y & -f_0^2 & xy' & yy' & f_0 y' \end{pmatrix}^\top, \\ \boldsymbol{\xi}^{(2)} &= \begin{pmatrix} f_0 x & f_0 y & f_0^2 & 0 & 0 & 0 & -xx' & -yx' & -f_0 x' \end{pmatrix}^\top, \\ \boldsymbol{\xi}^{(3)} &= \begin{pmatrix} -xy' & -yy' & -f_0 y' & xx' & yx' & f_0 x' & 0 & 0 & 0 \end{pmatrix}^\top, \end{aligned} \tag{5}$$

the three components of Eq. (4) are, after multiplication of f_0^2,

$$(\boldsymbol{\xi}^{(1)}, \boldsymbol{h}) = 0, \quad (\boldsymbol{\xi}^{(2)}, \boldsymbol{h}) = 0, \quad (\boldsymbol{\xi}^{(3)}, \boldsymbol{h}) = 0, \tag{6}$$

where $\boldsymbol{h}$ is a 9-D vector with components h_{11}, h_{12}, ..., h_{99}. Throughout this paper, we denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $(\boldsymbol{a}, \boldsymbol{b})$.

3. LS estimators

Let $\boldsymbol{\xi}_\alpha^{(k)}$ be the value of $\boldsymbol{\xi}^{(k)}$, $k = 1, 2, 3$, for $\{(x_\alpha, y_\alpha), (x'_\alpha, y'_\alpha)\}$, $\alpha = 1, ..., N$. Our task is to estimate an $\boldsymbol{h}$ such that $(\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{h}) \approx 0$, $k = 1, 2, 3$, $\alpha = 1, ..., N$. An *LS estimator* is the value of $\boldsymbol{h}$ that minimizes the sum of squares of the constraint equations, also known as the *algebraic distance*,

$$\begin{aligned} J &= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{3}(\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{h})^2 = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{3}\boldsymbol{h}^\top \boldsymbol{\xi}_\alpha^{(k)} \boldsymbol{\xi}_\alpha^{(k)\top} \boldsymbol{h} \\ &= (\boldsymbol{h}, \boldsymbol{M}\boldsymbol{h}), \end{aligned} \tag{7}$$

where we define the 9×9 matrix $\boldsymbol{M}$ by

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{3}\boldsymbol{\xi}_\alpha^{(k)} \boldsymbol{\xi}_\alpha^{(k)\top}. \tag{8}$$

Evidently, we need scale normalization on $\boldsymbol{h}$; otherwise, Eq. (7) is minimized by $\boldsymbol{h} = \boldsymbol{0}$. A frequently used convention is $h_{33} = 1$. Also, $\sum_{i,j=1}^{3} h_{ij}^2 = 1$ is widely used. However, the crucial fact is that *the value depends on the normalization*. Al-Sharadqah and Chernov [1], Rangarajan and Kanatani [13], and Kanatani and Rangarajan [8] exploited this freedom for circle/ellipse fitting and "optimized" the normalization so that the resulting estimator has high accuracy. In this paper, we do this for homography estimation.

Following [1, 8, 13], we consider the class of normalizations in the form

$$(\boldsymbol{h}, \boldsymbol{N}\boldsymbol{h}) = \text{constant}, \tag{9}$$

for some 9×9 symmetric matrix $\boldsymbol{N}$. If we let $\boldsymbol{N} = \boldsymbol{I}$ (unit matrix), we are requiring $\|\boldsymbol{h}\| = \text{constant}$. We call this the "standard LS". If $\boldsymbol{N}$ is positive definite, Eq. (9) is positive, so no generality is lost by setting it to 1. Like [1, 8, 13], however, we do not restrict $\boldsymbol{N}$ to be positive definite. As is well known, the solution $\boldsymbol{h}$ that minimizes Eq. (7) subject to Eq. (9) is obtained by solving the generalized eigenvalue problem

$$\boldsymbol{M}\boldsymbol{h} = \lambda \boldsymbol{N}\boldsymbol{h}. \tag{10}$$

The solution $\boldsymbol{h}$ has scale indeterminacy, so we normalize it to $\|\boldsymbol{h}\| = 1$ rather than Eq. (9). Our task is to select an appropriate $\boldsymbol{N}$ that gives the best solution $\boldsymbol{h}$, applying the perturbation theory of Kanatani [6] to Eq. (10).

4. Error Analysis

We assume that the observed positions (x_α, y_α) and (x'_α, y'_α) are perturbations of their true values $(\bar{x}_\alpha, \bar{y}_\alpha)$ and $(\bar{x}'_\alpha, \bar{y}'_\alpha)$ by independent Gaussian noise Δx_α, Δy_α, $\Delta x'_\alpha$, and $\Delta y'_\alpha$ of expectation 0 and standard deviation σ (pixels). The error terms $\Delta\boldsymbol{\xi}_\alpha^{(k)}$ of $\boldsymbol{\xi}_\alpha^{(k)}$ are

$$\Delta\boldsymbol{\xi}_\alpha^{(k)} = \Delta_1\boldsymbol{\xi}_\alpha^{(k)} + \Delta_2\boldsymbol{\xi}_\alpha^{(k)}, \tag{11}$$

where Δ_1 and Δ_2 denote, respectively, terms of orders 1 and 2 in Δx_α, Δy_α, $\Delta x'_\alpha$, and $\Delta y'_\alpha$. If we define the 9×4 Jacobi matrices $\boldsymbol{T}_\alpha^{(k)}$ of $\boldsymbol{\xi}_\alpha^{(k)}$ by

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

$$\boldsymbol{T}_\alpha^{(3)} = \begin{pmatrix} -\bar{y}'_\alpha & 0 & 0 & -\bar{x}_\alpha \\ 0 & -\bar{y}'_\alpha & 0 & -\bar{y}_\alpha \\ 0 & 0 & 0 & -f_0 \\ \bar{x}'_\alpha & 0 & \bar{x}_\alpha & 0 \\ 0 & \bar{x}'_\alpha & \bar{y}_\alpha & 0 \\ 0 & 0 & f_0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{12}$$

the first order terms $\Delta_1\boldsymbol{\xi}_\alpha^{(k)}$ are written as

$$\Delta_1\boldsymbol{\xi}_\alpha^{(k)} = \boldsymbol{T}_\alpha^{(k)} \begin{pmatrix} \Delta x_\alpha \\ \Delta y_\alpha \\ \Delta x'_\alpha \\ \Delta y'_\alpha \end{pmatrix}. \tag{13}$$

Using this, we define the covariance matrices of $\boldsymbol{\xi}_\alpha^{(k)}$ by

$$E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(l)}] = \boldsymbol{T}_\alpha^{(k)}E\Big[\begin{pmatrix}\Delta x_\alpha^2 & \Delta x_\alpha\Delta y_\alpha & \Delta x_\alpha\Delta x'_\alpha & \Delta x_\alpha\Delta y'_\alpha \\ \Delta y_\alpha\Delta x_\alpha & \Delta y_\alpha^2 & \Delta y_\alpha\Delta x'_\alpha & \Delta y_\alpha\Delta y'_\alpha \\ \Delta x'_\alpha\Delta x_\alpha & \Delta x'_\alpha\Delta y_\alpha & \Delta x'^2_\alpha & \Delta x'_\alpha\Delta y'_\alpha \\ \Delta y'_\alpha\Delta x_\alpha & \Delta y'_\alpha\Delta y_\alpha & \Delta y'_\alpha\Delta x'_\alpha & \Delta y'^2_\alpha\end{pmatrix}\Big]\boldsymbol{T}_\alpha^{(l)\top}$$
$$= \boldsymbol{T}_\alpha^{(k)}(\sigma^2\boldsymbol{I})\boldsymbol{T}_\alpha^{(l)\top} = \sigma^2\boldsymbol{T}_\alpha^{(k)}\boldsymbol{T}_\alpha^{(l)\top} = \sigma^2 V_0^{(kl)}[\boldsymbol{\xi}_\alpha], \quad (14)$$

where $E[\,\cdot\,]$ denotes expectation and we put

$$V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \equiv \boldsymbol{T}_\alpha^{(k)}\boldsymbol{T}_\alpha^{(l)\top}. \quad (15)$$

The second order error terms $\Delta_2\boldsymbol{\xi}_\alpha^{(k)}$ are given by

$$\Delta_2\boldsymbol{\xi}_\alpha^{(1)} = \big(0\ 0\ 0\ 0\ 0\ 0\ \Delta x_\alpha\Delta y'_\alpha\ \Delta y_\alpha\Delta y'_\alpha\ 0\big)^\top,$$
$$\Delta_2\boldsymbol{\xi}_\alpha^{(2)} = \big(0\ 0\ 0\ 0\ 0\ 0\ -\Delta x'_\alpha\Delta x_\alpha\ -\Delta x'_\alpha\Delta y_\alpha\ 0\big)^\top,$$
$$\Delta_2\boldsymbol{\xi}_\alpha^{(3)} = \big(-\Delta y'_\alpha\Delta x_\alpha\ -\Delta y'_\alpha\Delta y_\alpha\ 0\ \Delta x'_\alpha\Delta x_\alpha\ \Delta x'_\alpha\Delta y_\alpha\ 0\ 0\ 0\ 0\big)^\top. \quad (16)$$

5. Perturbation Analysis

Substituting Eq. (11) into Eq. (8), we obtain

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^N\sum_{k=1}^3(\bar{\boldsymbol{\xi}}_\alpha^{(k)}+\Delta_1\boldsymbol{\xi}_\alpha^{(k)}+\Delta_2\boldsymbol{\xi}_\alpha^{(k)})(\bar{\boldsymbol{\xi}}_\alpha^{(k)}+\Delta_1\boldsymbol{\xi}_\alpha^{(k)}+\Delta_2\boldsymbol{\xi}_\alpha^{(k)})^\top = \bar{\boldsymbol{M}} + \Delta_1\boldsymbol{M} + \Delta_2\boldsymbol{M} + \cdots, \quad (17)$$

where $\bar{\boldsymbol{M}}$ is the noise-free term, and $\cdots$ denotes terms of order 3 or higher in noise. The first and second order terms $\Delta_1\boldsymbol{M}$ and $\Delta_2\boldsymbol{M}$ are

$$\Delta_1\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^N\sum_{k=1}^3(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top} + \Delta_1\boldsymbol{\xi}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(k)\top}), \quad (18)$$

$$\Delta_2\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^N\sum_{k=1}^3(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_2\boldsymbol{\xi}_\alpha^{(k)\top} + \Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top} + \Delta_2\boldsymbol{\xi}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(k)\top}). \quad (19)$$

Accordingly, we expand $\boldsymbol{h}$ and λ in Eq. (10) in the form

$$\boldsymbol{h}=\bar{\boldsymbol{h}}+\Delta_1\boldsymbol{h}+\Delta_2\boldsymbol{h}+\cdots, \quad \lambda=\bar{\lambda}+\Delta_1\lambda+\Delta_2\lambda+\cdots. \quad (20)$$

Substituting Eqs. (17) and (20) into Eq. (10), we have

$$(\bar{\boldsymbol{M}}+\Delta_1\boldsymbol{M}+\Delta_2\boldsymbol{M}+\cdots)(\bar{\boldsymbol{h}}+\Delta_1\boldsymbol{h}+\Delta_2\boldsymbol{h}+\cdots) = (\bar{\lambda}+\Delta_1\lambda+\Delta_2\lambda+\cdots)\boldsymbol{N}(\bar{\boldsymbol{h}}+\Delta_1\boldsymbol{h}+\Delta_2\boldsymbol{h}+\cdots). \quad (21)$$

Equating terms of equal degrees in noise, we obtain

$$\bar{\boldsymbol{M}}\bar{\boldsymbol{h}} = \bar{\lambda}\boldsymbol{N}\bar{\boldsymbol{h}}, \quad (22)$$

$$\bar{\boldsymbol{M}}\Delta_1\boldsymbol{h} + \Delta_1\boldsymbol{M}\bar{\boldsymbol{h}} = \bar{\lambda}\boldsymbol{N}\Delta_1\boldsymbol{h} + \Delta_1\lambda\boldsymbol{N}\bar{\boldsymbol{h}}, \quad (23)$$

$$\bar{\boldsymbol{M}}\Delta_2\boldsymbol{h} + \Delta_1\boldsymbol{M}\Delta_1\boldsymbol{h} + \Delta_2\boldsymbol{M}\bar{\boldsymbol{h}} = \bar{\lambda}\boldsymbol{N}\Delta_2\boldsymbol{h} + \Delta_1\lambda\boldsymbol{N}\Delta_1\boldsymbol{h} + \Delta_2\lambda\boldsymbol{N}\bar{\boldsymbol{h}}. \quad (24)$$

Since $(\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{h}}) = 0$ for noise-free data, we have $\bar{\boldsymbol{M}}\bar{\boldsymbol{h}} = \boldsymbol{0}$ and hence $\bar{\lambda} = 0$ from Eq. (22). We see from Eq. (18) that $(\bar{\boldsymbol{h}}, \Delta_1\boldsymbol{M}\bar{\boldsymbol{h}}) = 0$. Computing the inner product of $\bar{\boldsymbol{h}}$ and Eq. (23), we see that $\Delta_1\lambda = 0$. Multiplying Eq. (23) by the pseudoinverse $\bar{\boldsymbol{M}}^-$ from left, we obtain

$$\Delta_1\boldsymbol{h} = -\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{h}}, \quad (25)$$

where we have noted that $\bar{\boldsymbol{h}}$ is a null vector of $\bar{\boldsymbol{M}}$ and hence $\boldsymbol{P}_{\bar{\boldsymbol{h}}} \equiv \bar{\boldsymbol{M}}^-\bar{\boldsymbol{M}}$ is the projection matrix in the direction of $\bar{\boldsymbol{h}}$. We have also noted that $\Delta_1\boldsymbol{h}$ is orthogonal to $\bar{\boldsymbol{h}}$ and hence $\boldsymbol{P}_{\bar{\boldsymbol{h}}}\Delta_1\boldsymbol{h} = \Delta_1\boldsymbol{h}$; this is easily seen by picking out first order terms from $\|\bar{\boldsymbol{h}} + \Delta_1\boldsymbol{h} + \Delta_2\boldsymbol{h} + \cdots\|^2 = 1$ [6].

Substituting Eq. (25) into Eq. (24), we see that $\Delta_2\lambda$ is given by

$$\Delta_2\lambda = \frac{(\bar{\boldsymbol{h}},\Delta_2\boldsymbol{M}\bar{\boldsymbol{h}}) - (\bar{\boldsymbol{h}},\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{h}})}{(\bar{\boldsymbol{h}},\boldsymbol{N}\bar{\boldsymbol{h}})} = \frac{(\bar{\boldsymbol{h}},\boldsymbol{T}\bar{\boldsymbol{h}})}{(\bar{\boldsymbol{h}},\boldsymbol{N}\bar{\boldsymbol{h}})}, \quad (26)$$

where we define

$$\boldsymbol{T} \equiv \Delta_2\boldsymbol{M} - \Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}. \quad (27)$$

Next, we consider the second order error $\Delta_2\boldsymbol{h}$. Since the magnitude of $\boldsymbol{h}$ is fixed to 1, we are only interested in the component orthogonal to $\bar{\boldsymbol{h}}$, which we denote by

$$\Delta_2^\perp\boldsymbol{h} = \boldsymbol{P}_{\bar{\boldsymbol{h}}}\Delta_2\boldsymbol{h}\ (= \bar{\boldsymbol{M}}^-\bar{\boldsymbol{M}}\Delta_2\boldsymbol{h}). \quad (28)$$

Multiplying Eq. (24) by $\bar{\boldsymbol{M}}^-$ from left and substituting Eq. (25), we obtain

$$\Delta_2^\perp\boldsymbol{h} = \Delta_2\lambda\bar{\boldsymbol{M}}^-\boldsymbol{N}\bar{\boldsymbol{h}} + \bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{h}} - \bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{M}\bar{\boldsymbol{h}} = \frac{(\bar{\boldsymbol{h}},\boldsymbol{T}\bar{\boldsymbol{h}})}{(\bar{\boldsymbol{h}},\boldsymbol{N}\bar{\boldsymbol{h}})}\bar{\boldsymbol{M}}^-\boldsymbol{N}\bar{\boldsymbol{h}} - \bar{\boldsymbol{M}}^-\boldsymbol{T}\bar{\boldsymbol{h}}. \quad (29)$$

6. Covariance and Bias

From Eq. (25), the leading term of the covariance matrix of the solution $\boldsymbol{h}$ is given by

$$V[\boldsymbol{h}] = E[\Delta_1\boldsymbol{h}\Delta_1\boldsymbol{h}^\top] = \frac{1}{N^2}\bar{\boldsymbol{M}}^-E[(\Delta_1\boldsymbol{M}\boldsymbol{h})(\Delta_1\boldsymbol{M}\boldsymbol{h})^\top]\bar{\boldsymbol{M}}^-$$
$$= \frac{1}{N^2}\bar{\boldsymbol{M}}^-E\Big[\sum_{\alpha=1}^N\sum_{k=1}^3(\Delta\boldsymbol{\xi}_\alpha^{(k)},\boldsymbol{h})\bar{\boldsymbol{\xi}}_\alpha^{(k)}\sum_{\beta=1}^N\sum_{l=1}^3(\Delta\boldsymbol{\xi}_\beta^{(l)},\boldsymbol{h})\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\Big]\bar{\boldsymbol{M}}^-$$
$$= \frac{1}{N^2}\bar{\boldsymbol{M}}^-\sum_{\alpha,\beta=1}^N\sum_{k,l=1}^3(\boldsymbol{h},E[\Delta\boldsymbol{\xi}_\alpha^{(k)}\Delta\boldsymbol{\xi}_\beta^{(l)\top}]\boldsymbol{h})\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\bar{\boldsymbol{M}}^-$$
$$= \frac{\sigma^2}{N^2}\bar{\boldsymbol{M}}^-\Big(\sum_{\alpha=1}^N\sum_{k,l=1}^3(\boldsymbol{h},V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\boldsymbol{h})\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}\Big)\bar{\boldsymbol{M}}^-$$
$$= \frac{\sigma^2}{N}\bar{\boldsymbol{M}}^-\bar{\boldsymbol{M}}'\bar{\boldsymbol{M}}^-, \quad (30)$$

where we define

$$\bar{\boldsymbol{M}}' = \frac{1}{N}\sum_{\alpha=1}^N\sum_{k,l=1}^3(\bar{\boldsymbol{h}},V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\boldsymbol{h})\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}. \quad (31)$$

In the above derivation, we have used our assumption that the noise in $\boldsymbol{\xi}_\alpha$ is independent for each α and that $E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}] = \delta_{\alpha\beta}\sigma^2 V_0^{(kl)}[\boldsymbol{\xi}_\alpha]$, where $\delta_{\alpha\beta}$ is the Kronecker delta. The important observation is that $V[\boldsymbol{h}]$ does not depend on the normalization weight $\boldsymbol{N}$. So, *all LS estimators have the same covariance matrix in the leading order*. Thus, we are unable to reduce the covariance of $\boldsymbol{h}$ by adjusting $\boldsymbol{N}$. This leads us to focus on the bias.

Since $E[\Delta_1\boldsymbol{h}] = \boldsymbol{0}$, the leading bias is $E[\Delta_2^\perp\boldsymbol{h}]$. To evaluate this, we first compute the expectation $E[\boldsymbol{T}]$ of $\boldsymbol{T}$ in Eq. (27). From Eq. (19), $E[\Delta_2\boldsymbol{M}]$ becomes

$$E[\Delta_2\boldsymbol{M}] = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\alpha^{(k)}E[\Delta_2\boldsymbol{\xi}_\alpha^{(k)}]^\top + E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}] + E[\Delta_2\boldsymbol{\xi}_\alpha^{(k)}]\bar{\boldsymbol{\xi}}_\alpha^{(k)\top}\Big) = \frac{\sigma^2}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{3}V_0^{(kk)}[\boldsymbol{\xi}_\alpha] = \sigma^2\boldsymbol{N}_{\rm T}, \tag{32}$$

where we put

$$\boldsymbol{N}_{\rm T} = \frac{1}{N}\sum_{k=1}^{3}V_0^{(kk)}[\boldsymbol{\xi}_\alpha]. \tag{33}$$

The term $E[\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}]$ is evaluated as follows (see Appendix A for the derivation):

$$E[\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}] = \frac{\sigma^2}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big({\rm tr}[\bar{\boldsymbol{M}}^-V_0^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} + (\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V_0^{(kl)}[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big). \tag{34}$$

Here, ${\rm tr}[\,\cdot\,]$ denotes the trace, and $\mathcal{S}[\,\cdot\,]$ means symmetrization ($\mathcal{S}[\boldsymbol{A}] = (\boldsymbol{A}+\boldsymbol{A}^\top)/2$). From Eqs. (32) and (34), the expectation of $\boldsymbol{T}$ in Eq. (27) is written as

$$E[\boldsymbol{T}] = \sigma^2\Big(\boldsymbol{N}_{\rm T} - \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big({\rm tr}[\bar{\boldsymbol{M}}^-V_0^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} + (\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V_0^{(kl)}[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big)\Big). \tag{35}$$

Hence, the expectation of $\Delta_2^\perp\boldsymbol{h}$ in Eq. (29) is

$$E[\Delta_2^\perp\boldsymbol{h}] = \bar{\boldsymbol{M}}^-\Big(\frac{(\bar{\boldsymbol{h}}, E[\boldsymbol{T}]\bar{\boldsymbol{h}})}{(\bar{\boldsymbol{h}}, \boldsymbol{N}\bar{\boldsymbol{h}})}\boldsymbol{N}\bar{\boldsymbol{h}} - E[\boldsymbol{T}]\bar{\boldsymbol{h}}\Big). \tag{36}$$

7. HyperLS estimator

Careful observation of Eqs. (35) and (36) reveals that if we choose $\boldsymbol{N}$ to be

$$\boldsymbol{N} = \boldsymbol{N}_{\rm T} - \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big({\rm tr}[\bar{\boldsymbol{M}}^-V_0^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} + (\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V_0^{(kl)}[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big), \tag{37}$$

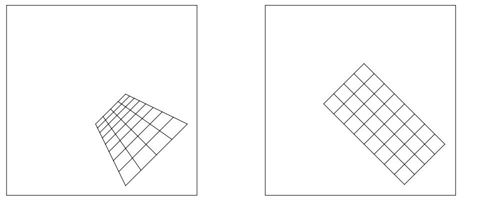

Figure 2 Simulated images of a planar surface.

then $E[\boldsymbol{T}] = \sigma^2\boldsymbol{N}$ from Eq. (35), and hence Eq. (36) becomes

$$E[\Delta_2^\perp\boldsymbol{h}] = \sigma^2\bar{\boldsymbol{M}}^-\Big(\frac{(\bar{\boldsymbol{h}}, \boldsymbol{N}\bar{\boldsymbol{h}})}{(\bar{\boldsymbol{h}}, \boldsymbol{N}\bar{\boldsymbol{h}})}\boldsymbol{N} - \boldsymbol{N}\Big)\bar{\boldsymbol{h}} = \boldsymbol{0}. \tag{38}$$

Since Eq. (37) contains the true values $\bar{\boldsymbol{\xi}}_\alpha^{(k)}$ and $\bar{\boldsymbol{M}}$, we evaluate them by replacing the true values $(\bar{x}_\alpha, \bar{y}_\alpha)$ and $(\bar{x}'_\alpha, \bar{y}'_\alpha)$ in their definitions by the observations (x_α, y_α) and (x'_α, y'_α), respectively. This does not affect the result, because expectations of odd-order error terms vanish and hence the error in Eq. (38) is at most $O(\sigma^4)$. Thus, the second order bias is *exactly* 0. After Al-Sharadqah and Chernov [1], Rangarajan and Kanatani [13], and Kanatani and Rangarajan [8], we call this *hyperLS*.

Standard linear algebra routines for solving generalized eigenvalue problems in the form of Eq. (10) assume that $\boldsymbol{N}$ is positive definite, but the matrix $\boldsymbol{N}$ in Eq. (37) is not guaranteed to be positive definite. However, this poses no problem, as Eq. (10) can be rewritten as

$$\boldsymbol{N}\boldsymbol{h} = (1/\lambda)\boldsymbol{M}\boldsymbol{h}. \tag{39}$$

Since the matrix $\boldsymbol{M}$ in Eq. (8) is positive definite for noisy data, we can solve Eq. (39) instead of Eq. (10). If the smallest eigenvalue of $\boldsymbol{M}$ happens to be 0, it indicates that the data are all exact; any method, e.g., the standard LS, gives an exact solution. The perturbation analysis of Kanatani [6] is based on the assumption that $\lambda \approx 0$, so we compute the unit generalized eigenvector for λ with the smallest absolute value.

The second term on the right-hand side of Eq. (37) is $O(1/N)$ and hence is expected to be small when N is large. We call the omission *Taubin approximation.*

8. Experiments

Figure 2 shows simulated images of a planar surface viewed from different directions. The image size is assumed to be 800×800 pixels with focal length f = 600 pixels. We added independent Gaussian noise of mean 0 and standard deviation σ (pixels) to the x and y coordinates of the grid points and computed the homography $\boldsymbol{h}$ from them. We measured the error of the computation by

$$\Delta^\perp\boldsymbol{h} = \boldsymbol{P}_{\bar{\boldsymbol{h}}}\hat{\boldsymbol{h}}, \qquad \boldsymbol{P}_{\bar{\boldsymbol{h}}} \equiv \boldsymbol{I} - \bar{\boldsymbol{h}}\bar{\boldsymbol{h}}^\top, \tag{40}$$

where $\hat{\boldsymbol{h}}$ and $\bar{\boldsymbol{h}}$ are the computed and the true values, respectively, and $\boldsymbol{P}_{\bar{\boldsymbol{h}}}$ is the projection matrix onto the direction orthogonal to $\bar{\boldsymbol{h}}$; we are only interested in

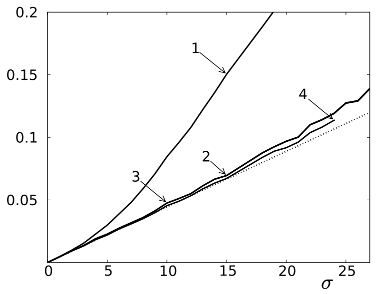

Figure 3 RMS error of the computed homography vs. the standard deviation σ of the added noise. 1. Standard LS. 2. HyperLS. 3. Taubin approximation. 4. ML. The halfway termination of the ML plot means that it did not converge beyond that noise level. The dotted line indicates the KCR lower bound.

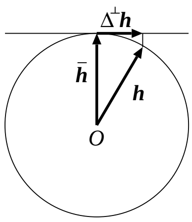

Figure 4 The error component $\Delta^\perp h$ of the computed value $\hat{h}$ orthogonal to the true value $\bar{h}$.

the error of $\hat{\boldsymbol{h}}$, which is a unit vector, orthogonal to $\bar{\boldsymbol{h}}$ (Fig. 4). For each σ, we evaluated the root-mean-square (RMS) error E of $\Delta^\perp \boldsymbol{h}$ over 1000 independent trials,

$$E = \sqrt{\frac{1}{1000}\sum_{a=1}^{1000} \|\Delta^\perp \boldsymbol{h}^{(a)}\|^2}, \qquad (41)$$

where the superscript (a) indicates the ath value. Figure 3 plots, for σ on the horizontal axis, the RMS error E of different methods: 1. standard LS, 2. hyperLS, 3. Taubin approximation, and 4. ML, for which we derived a new method by extending the FNS of Chojnacki [2] (see Appendix B). The dotted line shows the KCR lower bound [4, 5, 6]. The interrupted plot of ML means that the iterations failed to converge for σ larger than that (we used the standard LS to start the ML iterations).

We can see from Fig. 3 that the standard LS performs very poorly. In contrast, our hyperLS and its Taubin approximation almost compare with ML. Being algebraic, they do not fail for whatever noise. The accuracy of our hyperLS and ML (if it converges) are both close to the KCR lower bound. In practice, our hyperLS can be used to initialize ML iterations. In fact, we have observed that using our hyperLS, rather than the standard LS, to start the ML iterations in the experiment of Fig. 3 considerably extends the convergence range.

9. Conclusions

We presented a highly accurate LS alternative to the theoretically optimal ML estimator for homographies. Unlike ML, our hyperLS and its Taubin approximation are non-iterative and yield solutions even in the presence of large noise where ML computation may fail. Our approach is to adjust the normalization weight $\boldsymbol{N}$ so that the solution is unbiased up to second order noise terms. By numerical simulation, we demonstrated that our hyperLS outperforms the standard LS and has nearly comparable accuracy to ML, hence is suitable for initializing ML iterations.

Acknowledgments. This work was supported in part by the Ministry of Education, Culture, Sports, Science, and Technology, Japan, under a Grant in Aid for Scientific Research (C 21500172).

References

[1] A. Al-Sharadqah and N. Chernov: Error analysis for circle fitting algorithms, *Electronic J. Statistics*, **3** (2009-8), 886–911.

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

[3] R. Hartley and A. Zisserman: *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, Cambridge, U.K., 2004.

[4] K. Kanatani: *Statistical Optimization for Geometric Computation: Theory and Practice* Elsevier, Amsterdam, the Netherlands, 1996; reprinted, Dover, York, NY, U.S.A., 2005.

[5] K. Kanatani: Cramer-Rao lower bounds for curve fitting, *Graphical Models Image Process.*, **60**-2 (1998-03), 93–99.

[6] K. Kanatani: Statistical optimization for geometric fitting: Theoretical accuracy analysis and high order error analysis, *Int. J. Comput. Vis.*, **80**-2 (2008-11), 167–188.

[7] K. Kanatani: Unified computation of strict maximum likelihood for geometric fitting, *Proc. IS&T/SPIE 21st Annual Symp.*, Vol. 7239-EI103, San Jose, CA, U.S.A., January 2009, 72390Q-1 – 72390Q-12.

[8] K. Kanatani and P. Rangarajan, Hyperaccurate algebraic ellipse fitting without iterations, Proc. 5th Int. Conf. Comput. Vis. Theory Appl. Angers, France, May 2010, to appear.

[9] K. Kanatani and Y. Sugaya: Performance evaluation of iterative geometric fitting algorithms, *Comput. Stat. Data Anal.*, **52**-2 (2007-10), 1208–1222.

[10] K. Kanatani and Y. Sugaya: Compact fundamental matrix computation, *Proc. 2009 IEEE Pacific-Rim Symp. Image and Video Technology*, Tokyo, Japan, January 2009, 179–190.

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

[12] B. C. Matei and P. Meer: Estimation of nonlinear errors-in-variables models for computer vision applications, *IEEE Trans. Patt. Anal. Mach. Intell.*, **28**-10 (2006-10), 1537–1552.

[13] P. Rangarajan and K. Kanatani: Improved algebraic methods for circle fitting, *Elec. J. Stat.*, **3** (2009), 1075–1082.

[14] P. Rangarajan and P. Papamichalis: Estimating homographies without normalization, *Poc. Int. Conf. Image Process.*, Cairo, Egypt, November 2009, 3517–3520.

[15] T. Scoleri, W. Chojnacki, and M. J. Brooks: A multi-objective parameter estimator for image mosaicing, *Proc. 8th Int. Symp. Signal Processing and its Applications*, Sydney, Australia, Vol. 2, August 2005, 551–554.

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

Appendix

A. Derivation of Eq. (34)

The term $E[\Delta_1 \boldsymbol{M}\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M}]$ is computed as follows:

$$
\begin{aligned}
& E[\Delta_1 \boldsymbol{M}\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M}] \\
&= E[\frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}+\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(k)\top}\Big)\bar{\boldsymbol{M}}^- \\
&\quad \frac{1}{N}\sum_{\beta=1}^{N}\sum_{l=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}+\Delta_1\boldsymbol{\xi}_\beta^{(l)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\Big)] \\
&= \frac{1}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}E[(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}+\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(k)\top})\bar{\boldsymbol{M}}^- \\
&\quad (\bar{\boldsymbol{\xi}}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}+\Delta_1\boldsymbol{\xi}_\beta^{(l)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top})] \\
&= \frac{1}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}E[\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top} \\
&\quad +\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(k)\top}\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(k)\top}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}] \\
&= \frac{1}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}E[\bar{\boldsymbol{\xi}}_\alpha^{(k)}(\Delta_1\boldsymbol{\xi}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\bar{\boldsymbol{\xi}}_\alpha^{(k)}(\Delta_1\boldsymbol{\xi}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)})\bar{\boldsymbol{\xi}}_\beta^{(l)\top} \\
&\quad +\Delta_1\boldsymbol{\xi}_\alpha^{(k)}(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\Delta_1\boldsymbol{\xi}_\alpha^{(k)}(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)})\bar{\boldsymbol{\xi}}_\beta^{(l)\top}] \\
&= \frac{1}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}E[(\Delta_1\boldsymbol{\xi}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})\bar{\boldsymbol{\xi}}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +(\Delta_1\boldsymbol{\xi}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)})\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\Delta_1\boldsymbol{\xi}_\alpha^{(k)}(\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)},\bar{\boldsymbol{\xi}}_\alpha^{(k)})\bar{\boldsymbol{\xi}}_\beta^{(l)\top}] \\
&= \frac{1}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}E[\bar{\boldsymbol{\xi}}_\alpha^{(k)}((\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})^\top\Delta_1\boldsymbol{\xi}_\alpha^{(k)})\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\mathrm{tr}[\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{\xi}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top} \\
&\quad +\Delta_1\boldsymbol{\xi}_\alpha^{(k)}(\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)})\bar{\boldsymbol{\xi}}_\beta^{(l)\top}] \\
&= \frac{1}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\bar{\boldsymbol{M}}^-E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}] \\
&\quad +\mathrm{tr}[\bar{\boldsymbol{M}}^-E[\Delta_1\boldsymbol{\xi}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\alpha^{(k)\top}]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}] \\
&\quad +E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\Big) \\
&= \frac{\sigma^2}{N^2}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\bar{\boldsymbol{M}}^-\delta_{\alpha\beta}V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \\
&\quad +\mathrm{tr}[\bar{\boldsymbol{M}}^-\delta_{\alpha\beta}V_0^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta^{(l)})\delta_{\alpha\beta}V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \\
&\quad +\delta_{\alpha\beta}V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\Big) \\
&= \frac{\sigma^2}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}\bar{\boldsymbol{M}}^-V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \\
&\quad +\mathrm{tr}[\bar{\boldsymbol{M}}^-V_0^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \\
&\quad +V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}\Big) \\
&= \frac{\sigma^2}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big(\mathrm{tr}[\bar{\boldsymbol{M}}^-V_0^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)},\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V_0^{(kl)}[\boldsymbol{\xi}_\alpha] \\
&\quad +2\mathcal{S}[V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big). \qquad (42)
\end{aligned}
$$

Thus, Eq. (34) is obtained.

B. ML Homography Estimation

B.1 Formulation

If we assume that noise in $\boldsymbol{\xi}_\alpha^{(k)}$, $k = 1, 2, 3$, $\alpha = 1, ..., N$, is independent, isotropic, and Gaussian, then *maximum likelihood* (*ML*) of homography estimation reduces to minimizing the *Mahalanobis distance*, which equals the negative logarithm of the likelihood function up to a positive multiplicative constant and an additive constant. Thus, we minimize

$$
J_{\mathrm{ML}} = \frac{1}{N}\sum_{\alpha=1}^{N}\Big(\begin{pmatrix}\boldsymbol{\xi}_\alpha^{(1)}-\bar{\boldsymbol{\xi}}_\alpha^{(1)}\\ \boldsymbol{\xi}_\alpha^{(2)}-\bar{\boldsymbol{\xi}}_\alpha^{(2)}\\ \boldsymbol{\xi}_\alpha^{(3)}-\bar{\boldsymbol{\xi}}_\alpha^{(3)}\end{pmatrix},
\begin{pmatrix}V_0^{(11)}[\boldsymbol{\xi}_\alpha] & V_0^{(12)}[\boldsymbol{\xi}_\alpha] & V_0^{(13)}[\boldsymbol{\xi}_\alpha]\\ V_0^{(21)}[\boldsymbol{\xi}_\alpha] & V_0^{(22)}[\boldsymbol{\xi}_\alpha] & V_0^{(23)}[\boldsymbol{\xi}_\alpha]\\ V_0^{(31)}[\boldsymbol{\xi}_\alpha] & V_0^{(32)}[\boldsymbol{\xi}_\alpha] & V_0^{(33)}[\boldsymbol{\xi}_\alpha]\end{pmatrix}_4^-
\begin{pmatrix}\boldsymbol{\xi}_\alpha^{(1)}-\bar{\boldsymbol{\xi}}_\alpha^{(1)}\\ \boldsymbol{\xi}_\alpha^{(2)}-\bar{\boldsymbol{\xi}}_\alpha^{(2)}\\ \boldsymbol{\xi}_\alpha^{(3)}-\bar{\boldsymbol{\xi}}_\alpha^{(3)}\end{pmatrix}\Big), \qquad (43)
$$

where $V_0^{(kl)}[\boldsymbol{\xi}_\alpha]$ are the covariance matrices of $\Delta\boldsymbol{\xi}_\alpha^{(k)}$ in Eq. (15). The notation $(\cdot)_4^-$ denotes pseudoinverse of rank 4 with eigenvalues except the largest four being 0: The 27×27 matrix in Eq. (43) has rank 4 because the independent variables in $\Delta\boldsymbol{\xi}_\alpha^{(k)}$, $k = 1, 2, 3$, are only x_α, y_α, x'_α, and y'_α. We minimize Eq. (43) for $\boldsymbol{\xi}_\alpha^{(k)}$, $k = 1, 2, 3$, $\alpha = 1, ..., N$, and $\boldsymbol{h}$ subject to the constraint

$$
(\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \boldsymbol{h}) = 0. \qquad (44)
$$

The procedure for this computation was prescribed by Scoleri et al. [15], but their description is rather abstract, using Kronecker products and symbolic differentiations. Here, we evaluate all derivatives directly and write down all equations explicitly, using only standard arithmetics. This will more clearly reveal the underlying mathematical structure of the problem.

If we define 27-D vectors $\boldsymbol{\xi}_\alpha$ and 9×27 matrices $\boldsymbol{I}^{(1)}$, $\boldsymbol{I}^{(2)}$, and $\boldsymbol{I}^{(3)}$ by

$$\boldsymbol{\xi}_\alpha = \begin{pmatrix} \boldsymbol{\xi}_\alpha^{(1)} \\ \boldsymbol{\xi}_\alpha^{(2)} \\ \boldsymbol{\xi}_\alpha^{(3)} \end{pmatrix}, \tag{45}$$

$$\boldsymbol{I}^{(1)} = \begin{pmatrix} \boldsymbol{I} \\ \boldsymbol{O} \\ \boldsymbol{O} \end{pmatrix}, \quad \boldsymbol{I}^{(2)} = \begin{pmatrix} \boldsymbol{O} \\ \boldsymbol{I} \\ \boldsymbol{O} \end{pmatrix}, \quad \boldsymbol{I}^{(3)} = \begin{pmatrix} \boldsymbol{O} \\ \boldsymbol{O} \\ \boldsymbol{I} \end{pmatrix}, \tag{46}$$

where $\boldsymbol{I}$ is the 9×9 unit matrix, Eq. (44) is rewritten as

$$(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{I}^{(1)}\boldsymbol{h}) = 0, \quad (\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{I}^{(2)}\boldsymbol{h}) = 0, \quad (\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{I}^{(3)}\boldsymbol{h}) = 0. \tag{47}$$

Equation (43) is now rewritten as

$$J_{\rm ML} = \frac{1}{N}\sum_{\alpha=1}^{N}(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha]_4^-(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha)), \tag{48}$$

where $V_0[\boldsymbol{\xi}_\alpha]_4^-$ is the 27×27 matrix in Eq. (43). Introducing Lagrange multipliers $\lambda_\alpha^{(k)}$ to Eqs. (47), differentiating

$$\frac{1}{2}NJ_{\rm ML} - \sum_{k=1}^{3}\lambda_\alpha^{(k)}(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{I}^{(k)}\boldsymbol{h}), \tag{49}$$

with respect to $\bar{\boldsymbol{\xi}}_\alpha$, and setting the result to 0, we obtain

$$-V_0[\boldsymbol{\xi}_\alpha]_4^-(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha) - \sum_{k=1}^{3}\lambda_\alpha^{(k)}\boldsymbol{I}^{(k)}\boldsymbol{h} = \boldsymbol{0}. \tag{50}$$

Multiplying this with $V_0[\boldsymbol{\xi}_\alpha]$ from left, we have

$$-(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha) - \sum_{k=1}^{3}\lambda_\alpha^{(k)}V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{I}^{(k)}\boldsymbol{h} = \boldsymbol{0}, \tag{51}$$

where we have noted that the perturbation $\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha$ due to noise takes place within the domain of the covariance matrix $V_0[\boldsymbol{\xi}_\alpha]$ and hence is invariant to the projection $V_0[\boldsymbol{\xi}_\alpha]V_0[\boldsymbol{\xi}_\alpha]_4^-$ onto the domain of $V_0[\boldsymbol{\xi}_\alpha]$. Substituting the expression of $V_0[\boldsymbol{\xi}_\alpha]_4^-(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha)$ obtained from Eq. (50) and the expression of $\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha$ obtained from Eq. (51) into Eq. (48), we can write $J_{\rm ML}$ in the form

$$\begin{aligned}
J_{\rm ML} &= \frac{1}{N}\sum_{\alpha=1}^{N}(\sum_{k=1}^{3}\lambda_\alpha^{(k)}V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{I}^{(k)}\boldsymbol{h}, \sum_{l=1}^{3}\lambda_\alpha^{(l)}\boldsymbol{I}^{(l)}\boldsymbol{h}) \\
&= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\lambda_\alpha^{(k)}\lambda_\alpha^{(l)}(\boldsymbol{h}, \boldsymbol{I}^{(kl)\top}V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{I}^{(k)}\boldsymbol{h}) \\
&= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\lambda_\alpha^{(k)}\lambda_\alpha^{(l)}(\boldsymbol{h}, V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\boldsymbol{h}) \\
&= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\lambda_\alpha^{(k)}\lambda_\alpha^{(l)}V_\alpha^{(kl)},
\end{aligned} \tag{52}$$

where we put

$$V_\alpha^{(kl)} = (\boldsymbol{h}, V_0^{(kl)}[\boldsymbol{\xi}_\alpha]\boldsymbol{h}). \tag{53}$$

If we substitute the expression of $\bar{\boldsymbol{\xi}}_\alpha$ obtained from Eq. (51) into Eqs. (47), we have

$$\sum_{l=1}^{3}V_\alpha^{(kl)}\lambda_\alpha^{(l)} = -(\boldsymbol{\xi}_\alpha, \boldsymbol{I}^{(k)}\boldsymbol{h}) = -(\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{h}), \quad k = 1, 2, 3, \tag{54}$$

which provides simultaneous linear equations for $\lambda_\alpha^{(k)}$. However, the rank of the coefficient matrix $\boldsymbol{V}_\alpha = (V_\alpha^{(kl)})$ drops to 2 if there is no noise (as described shortly). So, we solve Eq. (54) by least squares, which is equivalent to using the pseudoinverse $\boldsymbol{W}_\alpha = (\boldsymbol{V}_\alpha)_2^-$ of rank 2, obtaining

$$\lambda_\alpha^{(k)} = -\sum_{l=1}^{3}W_\alpha^{(kl)}(\boldsymbol{\xi}_\alpha^{(l)}, \boldsymbol{h}). \tag{55}$$

Substituting this into Eq. (52), we obtain

$$\begin{aligned}
J_{\rm ML} &= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big(\sum_{m=1}^{3}W_\alpha^{(km)}(\boldsymbol{\xi}_\alpha^{(m)}, \boldsymbol{h})\Big) \\
&\quad \Big(\sum_{n=1}^{3}W_\alpha^{(ln)}(\boldsymbol{\xi}_\alpha^{(n)}, \boldsymbol{h})\Big)V_\alpha^{(kl)} \\
&= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{m,n=1}^{3}\Big(\sum_{k,l=1}^{3}W_\alpha^{(km)}V_\alpha^{(kl)}W_\alpha^{(ln)}\Big) \\
&\quad (\boldsymbol{\xi}_\alpha^{(m)}, \boldsymbol{h})(\boldsymbol{\xi}_\alpha^{(n)}, \boldsymbol{h}) \\
&= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{m,n=1}^{3}W_\alpha^{(mn)}(\boldsymbol{\xi}_\alpha^{(m)}, \boldsymbol{h})(\boldsymbol{\xi}_\alpha^{(n)}, \boldsymbol{h}),
\end{aligned} \tag{56}$$

where we have used the identity for pseudo inverse: $\boldsymbol{W}_\alpha\boldsymbol{V}_\alpha\boldsymbol{W}_\alpha = \boldsymbol{W}_\alpha(\boldsymbol{W}_\alpha)_2^-\boldsymbol{W}_\alpha = \boldsymbol{W}_\alpha$.

The expression of this type is called the *Sampson error*. Note that *no approximation has been introduced* to derive Eq. (56). However, we assumed in the beginning that noise in $\boldsymbol{\xi}_\alpha^{(k)}$ is Gaussian. This is not strictly true if Δx_α, Δy_α, $\Delta x'_\alpha$, and $\Delta y'_\alpha$ is Gaussian. It has been confirmed in many problems that the Gaussian approximation of noise in $\boldsymbol{\xi}_\alpha^{(k)}$, or the *Sampson approximation*, does practically not affect the solution of the strict ML solution [7].

B.2 Minimizing Eq. (56)

It is easily seen from the definition of $\boldsymbol{\xi}_\alpha^{(k)}$ that

$$x'_\alpha\boldsymbol{\xi}_\alpha^{(1)} + y'_\alpha\boldsymbol{\xi}_\alpha^{(2)} + f_0\boldsymbol{\xi}_\alpha^{(3)} = \boldsymbol{0} \tag{57}$$

holds identically. Computing the inner product with $\boldsymbol{h}$ on both sides, we obtain

$$(x'_\alpha\boldsymbol{\xi}_\alpha^{(1)} + y'_\alpha\boldsymbol{\xi}_\alpha^{(2)} + f_0\boldsymbol{\xi}_\alpha^{(3)}, \boldsymbol{h}) = 0. \tag{58}$$

This is an identity in x_α, y_α, x'_α, and y'_α, so its derivatives with respect to these are also identities. Hence, the following identically holds if there is no noise:

$$\begin{aligned}
(x'_\alpha[\boldsymbol{T}_\alpha^{(1)}]_1 + y'_\alpha[\boldsymbol{T}_\alpha^{(2)}]_1 + f_0[\boldsymbol{T}_\alpha^{(3)}]_1, \boldsymbol{h}) &= 0, \\
(x'_\alpha[\boldsymbol{T}_\alpha^{(1)}]_2 + y'_\alpha[\boldsymbol{T}_\alpha^{(2)}]_2 + f_0[\boldsymbol{T}_\alpha^{(3)}]_2, \boldsymbol{h}) &= 0, \\
(x'_\alpha[\boldsymbol{T}_\alpha^{(1)}]_3 + y'_\alpha[\boldsymbol{T}_\alpha^{(2)}]_3 + f_0[\boldsymbol{T}_\alpha^{(3)}]_3, \boldsymbol{h}) &= 0, \\
(x'_\alpha[\boldsymbol{T}_\alpha^{(1)}]_4 + y'_\alpha[\boldsymbol{T}_\alpha^{(2)}]_4 + f_0[\boldsymbol{T}_\alpha^{(3)}]_4, \boldsymbol{h}) &= 0.
\end{aligned} \tag{59}$$

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

$$(x'_\alpha \boldsymbol{T}_\alpha^{(1)} + y'_\alpha \boldsymbol{T}_\alpha^{(2)} + f_0 \boldsymbol{T}_\alpha^{(3)})^\top \boldsymbol{h} = \boldsymbol{0}. \tag{60}$$

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

$$(x'_\alpha V_0^{(k1)}[\boldsymbol{\xi}_\alpha] + y'_\alpha V_0^{(k2)}[\boldsymbol{\xi}_\alpha] + f_0 V_0^{(k3)}[\boldsymbol{\xi}_\alpha])\boldsymbol{h} = \boldsymbol{0}. \tag{61}$$

We 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. (61), we obtain

$$\boldsymbol{V}_\alpha \begin{pmatrix} x'_\alpha \\ y'_\alpha \\ f_0 \end{pmatrix} = \boldsymbol{0}. \tag{62}$$

Thus, $\boldsymbol{x}'_\alpha = \begin{pmatrix} x'_\alpha & y'_\alpha & f_0 \end{pmatrix}^\top$ 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 $\boldsymbol{x}'_\alpha$. Hence, we can write

$$\boldsymbol{W}_\alpha \boldsymbol{V}_\alpha = \boldsymbol{V}_\alpha \boldsymbol{W}_\alpha = \boldsymbol{I} - \mathcal{N}[\boldsymbol{x}'_\alpha]\mathcal{N}[\boldsymbol{x}'_\alpha]^\top, \tag{63}$$

where $\mathcal{N}[\,\cdot\,]$ denotes normalization into unit norm. Differentiating Eq. (63) with respect to h_i, we obtain

$$\frac{\partial \boldsymbol{V}_\alpha}{\partial h_i} \boldsymbol{W}_\alpha + \boldsymbol{V}_\alpha \frac{\partial \boldsymbol{W}_\alpha}{\partial h_i} = \boldsymbol{O}. \tag{64}$$

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

$$\frac{\partial \boldsymbol{W}_\alpha}{\partial h_i} = -\boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i} \boldsymbol{W}_\alpha. \tag{65}$$

Now, if we define the 9×3 matrix

$$\Xi_\alpha = \begin{pmatrix} \boldsymbol{\xi}_\alpha^{(1)} & \boldsymbol{\xi}_\alpha^{(2)} & \boldsymbol{\xi}_\alpha^{(3)} \end{pmatrix}, \tag{66}$$

Eq. (56) can be rewritten as follows:

$$J_{\mathrm{ML}} = \frac{1}{N} \sum_{\alpha=1}^{N} (\boldsymbol{h}, \Xi_\alpha \boldsymbol{W}_\alpha \Xi_\alpha^\top \boldsymbol{h}). \tag{67}$$

Differentiating this with respect to h_i and using Eq. (65), we obtain

$$\frac{\partial J_{\mathrm{ML}}}{\partial h_i} = \frac{2}{N} \sum_{\alpha=1}^{N} (\Xi_\alpha \boldsymbol{W}_\alpha \Xi_\alpha^\top \boldsymbol{h})_i - \frac{2}{N} \sum_{\alpha=1}^{N} (\boldsymbol{h}, \Xi_\alpha \boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i} \boldsymbol{W}_\alpha \Xi_\alpha^\top \boldsymbol{h}), \tag{68}$$

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

$$v_\alpha^{(k)} = \sum_{l=1}^{3} W_\alpha^{(kl)} (\boldsymbol{\xi}_\alpha^{(l)}, \boldsymbol{h}), \tag{69}$$

and define $\boldsymbol{v}_\alpha$ to be the 3-D vector with components $v_\alpha^{(k)}$, $k = 1, 2, 3$, Eq. (69) is written as

$$\boldsymbol{v}_\alpha = \boldsymbol{W}_\alpha \Xi_\alpha^\top \boldsymbol{h}. \tag{70}$$

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 $2\sum_{j=1}^{9} V_0^{(kl)}[\boldsymbol{\xi}_\alpha]_{ij} h_j$. Hence, the last term of the right-hand side of Eq. (68) is

$$\frac{2}{N}\sum_{\alpha=1}^{N}(\boldsymbol{h}, \Xi_\alpha \boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i} \boldsymbol{W}_\alpha \Xi_\alpha^\top \boldsymbol{h}) = \frac{2}{N}\sum_{\alpha=1}^{N}(\boldsymbol{v}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial h_i} \boldsymbol{v}_\alpha)$$
$$= \sum_{j=1}^{9} \Big(\frac{2}{N} \sum_{\alpha=1}^{N} \sum_{k,l=1}^{3} V_0^{(kl)}[\boldsymbol{\xi}_\alpha]_{ij} v_\alpha^{(k)} v_\alpha^{(l)} \Big) h_j. \tag{71}$$

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

$$\boldsymbol{M}_{\mathrm{ML}} = \frac{1}{N} \sum_{\alpha=1}^{N} W_\alpha^{(kl)} \boldsymbol{\xi}_\alpha^{(k)} \boldsymbol{\xi}_\alpha^{(l)\top}, \tag{72}$$

$$\boldsymbol{L}_{\mathrm{ML}} = \frac{1}{N} \sum_{\alpha=1}^{N} \sum_{k,l=1}^{N} v_\alpha^{(k)} v_\alpha^{(l)} V_0^{(kl)}[\boldsymbol{\xi}_\alpha], \tag{73}$$

the first term on the right-hand side of Eq. (68) is simply $2\boldsymbol{M}_{\mathrm{ML}}$. Equation (71) is written as $2\boldsymbol{L}_{\mathrm{ML}}\boldsymbol{h}$. Thus, we obtain the following expression of the derivative of J_{ML} in Eq. (67):

$$\nabla_{\boldsymbol{h}} J_{\mathrm{ML}} = 2(\boldsymbol{M}_{\mathrm{ML}} - \boldsymbol{L}_{\mathrm{ML}})\boldsymbol{h}. \tag{74}$$

It follows that to minimize J_{ML} we need to solve

$$(\boldsymbol{M}_{\mathrm{ML}} - \boldsymbol{L}_{\mathrm{ML}})\boldsymbol{h} = \boldsymbol{0}. \tag{75}$$

In the above derivation, we have assumed that there is no noise. In the presence of noise, the only difference is that Eq. (62) does not exactly hold, and $\boldsymbol{V}_\alpha$ is nonsingular with the smallest eigenvalue close to 0. So, we regard $\boldsymbol{W}_\alpha = (\boldsymbol{V}_\alpha)_2^-$ as obtained by curtailing the smallest eigenvalue of $\boldsymbol{V}_\alpha$ to 0.

B.3 Solving Eq. (75)

In order to solve Eq. (75), we use the FNS principle of Chojnacki et al. [2], though we may as well use the HEIV principle of Leedan and Meer [11] and Matei and Meer [12]. The FNS procedure goes as follows:

1. Provide an initial value $\boldsymbol{h}_0$ for $\boldsymbol{h}$ (e.g., by the standard LS).
2. Compute the matrices $\boldsymbol{M}_{\mathrm{ML}}$ and $\boldsymbol{L}_{\mathrm{ML}}$ in Eqs. (72) and (73).
3. Solve the eigenvalue problem
$$(\boldsymbol{M}_{\mathrm{ML}} - \boldsymbol{L}_{\mathrm{ML}})\boldsymbol{h} = \lambda \boldsymbol{h}, \tag{76}$$
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 $\boldsymbol{h}_0 \leftarrow \mathcal{N}[\boldsymbol{h}_0 + \boldsymbol{h}]$, and go back to Step 2.

The term $\mathcal{N}[\boldsymbol{h}_0 + \boldsymbol{h}]$ means $\mathcal{N}[(\boldsymbol{h}_0 + \boldsymbol{h})/2]$. This average taking, not originally shown by Chojnacki et al. [2], was shown to stabilize the convergence in many problems [10].