IPSJ Transactions on Computer Vision and Applications Vol. 3 80–94 (Oct. 2011)

*Regular Paper*

# HyperLS for Parameter Estimation in Geometric Fitting

Kenichi Kanatani,[†1] Prasanna Rangarajan,[†2]
Yasuyuki Sugaya[†3] and Hirotaka Niitsuma[†1]

We present a general framework of a special type of least squares (LS) estimator, which we call "HyperLS," for parametiper estimation that frequently arises in computer vision applications. It minimizes the algebraic distance under a special scale normalization, which is derived by a detailed error analysis in such a way that statistical bias is removed up to second order noise terms. We discuss in detail many theoretical issues involved in its derivation. By numerical experiments, we show that HyperLS is far superior to the standard LS and comparable in accuracy to maximum likelihood (ML), which is known to produce highly accurate results but may fail to converge if poorly initialized. We conclude that HyperLS is a perfect candidate for ML initialization.

## 1. Introduction

An important task in computer vision is the extraction of 2-D/3-D geometric information from image data[7),8)], for which we often need to estimate parameters from observations that should satisfy implicit polynomials in the absence of noise. For such a problem, maximum likelihood (ML) is known to produce highly accurate solutions, achieving the theoretical accuracy limit to a first approximation in the noise level[3),8),10)]. However, ML requires iterative search, which does not always converge unless started from a value sufficiently close to the solution. For this reason, various numerical schemes that can produce reasonably accurate approximations have been extensively studied[7)]. The simplest of such schemes is *algebraic distance minimization*, or simply *least squares* (LS), which minimizes the sum of the squares of polynomials that should be zero in the absence of noise. However, the accuracy of LS is very much limited. Recently, a new approach for increasing the accuracy of LS has been proposed in several applications[1),12),22)–24)]. In this paper, we call it *HyperLS* and present a unified formulation and clarify various theoretical issues that have not been fully studied so far.

Section 2 defines the mathematical framework of the problem with illustrating examples. Section 3 introduces a statistical model of observation. In Section 4, we discuss various issues of ML. Section 5 describes a general framework of algebraic fitting. In Sections 6 and 7, we do a detailed error analysis of algebraic fitting in general and in Section 8 derive expressions of covariance and bias of the solution. In Section 9, we define HyperLS by choosing the scale normalization that eliminates the bias up to second order noise terms. In Section 10, we do numerical experiments to show that HyperLS is far superior to the standard LS and is comparable in accuracy to ML, which implies that HyperLS is a perfect candidate for initializing the ML iterations. In Section 11, we conclude.

## 2. Geometric Fitting

The term "image data" in this paper refers to values extracted from images by image processing operations such as edge filters and interest point detectors. An example of image data includes the locations of points that have special characteristics in the images or the lines that separate image regions having different properties. We say that image data are "noisy" in the sense that image processing operations for detecting them entail uncertainty to some extent. Let $\boldsymbol{x}_1$, ... $\boldsymbol{x}_N$ be noisy image data, which we regard as perturbations in their true values $\bar{\boldsymbol{x}}_1$, ..., $\bar{\boldsymbol{x}}_N$ that satisfy implicit geometric constraints of the form

$$F^{(k)}(\boldsymbol{x};\boldsymbol{\theta}) = 0, \qquad k = 1, \ldots, L. \tag{1}$$

The unknown parameter $\boldsymbol{\theta}$ allows us to infer the 2-D/3-D shape and motion of the objects observed in the images[7),8)]. We call this type of problem *geometric fitting*[8)]. In many vision applications, we can reparameterize the problem to make the functions $F^{(k)}(\boldsymbol{x};\boldsymbol{\theta})$ linear in $\boldsymbol{\theta}$ (but generally nonlinear in $\boldsymbol{x}$), allowing us to write Eq. (1) as

$$(\boldsymbol{\xi}^{(k)}(\boldsymbol{x}), \boldsymbol{\theta}) = 0, \qquad k = 1, \ldots, L, \tag{2}$$

where and hereafter $(\boldsymbol{a}, \boldsymbol{b})$ denotes the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. The

[†1] Okayama University
[†2] Southern Methodist University
[†3] Toyohashi University of Technology

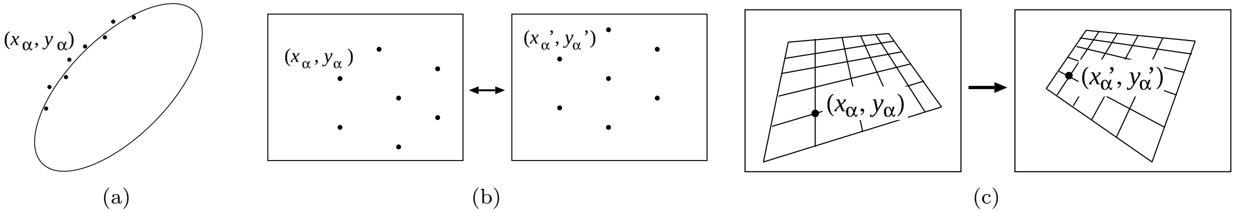


**Fig. 1** (a) Fitting an ellipse to a point sequence. (b) Computing the fundamental matrix from corresponding points between two images. (c) Computing a homography between two images.

vector $\boldsymbol{\xi}^{(k)}(\boldsymbol{x})$ represents a nonlinear mapping of $\boldsymbol{x}$.

**Example 1 (Ellipse fitting).** Given a point sequence $(x_\alpha, y_\alpha)$, $\alpha = 1, \ldots N$, we wish to fit an ellipse of the form

$$Ax^2 + 2Bxy + Cy^2 + 2(Dx + Ey) + F = 0. \tag{3}$$

(**Fig. 1**(a)). If we let

$$\boldsymbol{\xi} = (x^2, 2xy, y^2, 2x, 2y, 1)^\top, \qquad \boldsymbol{\theta} = (A, B, C, D, E, F)^\top, \tag{4}$$

the constraint in Eq. (3) has the form of Eq. (2) with $L = 1$.

**Example 2 (Fundamental matrix computation).** Corresponding points $(x, y)$ and $(x', y')$ in two images of the same 3-D scene taken from different positions satisfy the *epipolar equation*[7)]

$$(\boldsymbol{x}, \boldsymbol{F}\boldsymbol{x}') = 0, \qquad \boldsymbol{x} \equiv (x, y, 1)^\top, \qquad \boldsymbol{x}' \equiv (x', y', 1')^\top, \tag{5}$$

where $\boldsymbol{F}$ is called the *fundamental matrix*, from which we can compute the camera positions and the 3-D structure of the scene[7),8)] (Fig. 1(b)). If we let

$$\begin{aligned} \boldsymbol{\xi} &= (xx', xy', x, yx', yy', y, x', y', 1)^\top, \\ \boldsymbol{\theta} &= (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})^\top, \end{aligned} \tag{6}$$

the constraint in Eq. (5) has the form of Eq. (2) with $L = 1$.

**Example 3 (Homography computation).** Two images of a planar or infinitely far away scene are related by a *homography* of the form

$$\boldsymbol{x}' \simeq \boldsymbol{H}\boldsymbol{x}, \qquad \boldsymbol{x} \equiv (x, y, 1)^\top, \qquad \boldsymbol{x}' \equiv (x', y', 1')^\top, \tag{7}$$

where $\boldsymbol{H}$ is a nonsingular matrix, and $\simeq$ denotes equality up to a nonzero multiplier[7),8)] (Fig. 1(c)). We can alternatively express Eq. (7) as the vector product equality

$$\boldsymbol{x}' \times \boldsymbol{H}\boldsymbol{x} = \boldsymbol{0}. \tag{8}$$

If we let

$$\begin{aligned} \boldsymbol{\xi}^{(1)} &= (0, 0, 0, -x, -y, -1, xy', yy', y')^\top, \\ \boldsymbol{\xi}^{(2)} &= (x, y, 1, 0, 0, 0, -xx', -yx', -x')^\top, \\ \boldsymbol{\xi}^{(3)} &= (-xy', -yy', -y', xx', yx', x', 0, 0, 0)^\top, \end{aligned} \tag{9}$$

$$\boldsymbol{\theta} = (H_{11}, H_{12}, H_{13}, H_{21}, H_{22}, H_{23}, H_{31}, H_{32}, H_{33})^\top, \tag{10}$$

the three components of Eq. (8) have the form of Eq. (2) with $L = 3$. Note that $\boldsymbol{\xi}^{(1)}$, $\boldsymbol{\xi}^{(2)}$, and $\boldsymbol{\xi}^{(3)}$ in Eq. (9) are linearly dependent; only two of them are independent.

## 3. Statistical Model of Observation

Before proceeding to the error analysis of the above problems, we need to introduce a statistical model of observation. We regard each datum $\boldsymbol{x}_\alpha$ as perturbed from its true value $\bar{\boldsymbol{x}}_\alpha$ by $\Delta\boldsymbol{x}_\alpha$, which we assume to be independent Gaussian noise of mean $\boldsymbol{0}$ and covariance matrix $V[\boldsymbol{x}_\alpha]$. We do not impose any restrictions on the true values $\bar{\boldsymbol{x}}_\alpha$ except that they should satisfy Eq. (1). This is known as a *functional* model. We could alternatively introduce some statistical model according to which the true values $\bar{\boldsymbol{x}}_\alpha$ are sampled. Then, the model is called *structural*. This distinction is crucial when we consider limiting processes in the following sense[10)].

Conventional statistical analysis mainly focuses on the asymptotic behavior as the number of observations increases to $\infty$. This is based on the reasoning that the mechanism underlying noisy observations would better reveal itself as the number of observations increases (the law of large numbers) while the number of available data is limited in practice. So, the estimation accuracy vs. the number of data is a major concern. In this light, efforts have been made to obtain a *consistent* estimator for fitting an ellipse to noisy data or computing the fundamental matrix from noisy point correspondences such that the solution approaches its true value in the limit $N \to \infty$ of the number $N$ of the data[17),18)].

In image processing applications, in contrast, one cannot "repeat" observations. One makes an inference given a single set of images, and how many times one applies image processing operations, the result is always the same, because standard image processing algorithms are deterministic; no randomness is involved. This is in a stark contrast to conventional statistical problems, where we view observations as "samples" from potentially infinitely many possibilities and could obtain, by repeating observations, different values originating from unknown, uncontrollable, or unmodeled causes, which we call "noise" as a whole.

In image-based applications, the accuracy of inference deteriorates as the uncertainty of image processing operations increases. Thus, the inference accuracy vs. the uncertainty of image operations, which we call "noise" for simplicity, is a major concern. Usually, the noise is very small, often subpixel levels. In light of this observation, it has been pointed out that in image domains the "consistency" of estimators should more appropriately be defined by the behavior in the limit $\sigma \to 0$ of the noise level $\sigma$[3),10)].

In this paper, we are interested in image processing applications and focus on the perturbation analysis around $\sigma = 0$ with the number $N$ of data fixed. Thus, the functional model suits our purpose. If we want to analyze the error behavior in the limit of $N \to \infty$, we need to assume some structural model that specifies how the statistical characteristics of the data depend on $N$. The derivation of consistent estimators for $N \to \infty$ is based on such an assumption[17),18)]. However, it is difficult to predict the noise characteristics for different $N$. Image processing filters usually output a list of points or lines or their correspondences along with their confidence values, from which we use only those with high confidence. If we want to collect a lot of data, we necessarily need to include those with low confidence, but their statistical properties are hard to estimate, since such data are possibly misdetections. This is the most different aspect of image processing from laboratory experiments, in which any number of data can be collected by repeated trials.

## 4. Maximum Likelihood for Geometric Fitting

Under the Gaussian noise model, maximum likelihood (ML) of our problem can be written as the minimization of the Mahalanobis distance

$$I = \sum_{\alpha=1}^{N} (\bar{\boldsymbol{x}}_\alpha - \boldsymbol{x}_\alpha, V[\boldsymbol{x}_\alpha]^{-1}(\bar{\boldsymbol{x}}_\alpha - \boldsymbol{x}_\alpha)), \tag{11}$$

with respect to $\bar{\boldsymbol{x}}_\alpha$ subject to the constraint that

$$(\boldsymbol{\xi}^{(k)}(\bar{\boldsymbol{x}}_\alpha), \boldsymbol{\theta}) = 0, \qquad k = 1, \ldots, L, \tag{12}$$

for some $\boldsymbol{\theta}$. If the noise is homogeneous and isotropic, Eq. (11) is the sum of the squares of the geometric distances between the observations $\boldsymbol{x}_\alpha$ and their true values $\bar{\boldsymbol{x}}_\alpha$, often referred to as the *reprojection error*[7)]. That name originates from the following intuition: We infer the 3-D structure of the scene from its projected images, and when the inferred 3-D structure is "reprojected" onto the images, Eq. (11) measures the discrepancy between the "reprojections" of our solution and the actual observations.

In statisitcs, ML is criticized for its lack of consistency[17),18)]. In fact, estimation of the true values $\bar{\boldsymbol{x}}_\alpha$, called *nuisance parameters* when viewed as parameters, is not consistent as $N \to \infty$ in the ML framework, as pointed out by Neyman and Scott[21)] as early as in 1948. As discussed in the preceding section, however, the lack of consistency has no realistic meaning in vision applications. On the contrary, ML has very desirable properties in the limit $\sigma \to 0$ of the noise level $\sigma$: the solution is "consistent" in the sense that it converges to the true value as $\sigma \to 0$ and "efficient" in the sense that its covariance matrix approaches a theoretical lower bound as $\sigma \to 0$[3),8),10)].

According to the experience of many vision researchers, ML is known to produce highly accurate solutions[7)], and no necessity is felt for further accuracy improvement. Rather, a major concern is its computational burden, because ML usually requires complicated nonlinear optimization.

The standard approach is to introduce some auxiliary parameters to express each of $\bar{\boldsymbol{x}}_\alpha$ explicitly in terms of $\boldsymbol{\theta}$ and the auxiliary parameters. After they are substituted back into Eq. (11), the Mahalanobis distance $I$ becomes a function of $\boldsymbol{\theta}$ and the auxiliary parameters. Then, this joint parameter space, which usually has very high dimensions, is searched for the minimum. This approach is called *bundle adjustment*[7),27)], a term originally used by photogrammetrists. This is very time consuming, in particular if one seeks a globally optimal solution by searching the entire parameter space exhaustively[6)].

A popular alternative to bundle adjustment is minimization of a function of $\boldsymbol{\theta}$ alone, called the *Sampson error*[7)], which approximates the minimum of Eq. (11) for a given $\boldsymbol{\theta}$ (the actual expression is shown in Section 6). Kanatani and Sugaya[16)] showed that the exact ML solution can be obtained by repeating Sampson error minimization, each time modifying the Sampson error so that in the end the modified Sampson error coincides with the Mahalanobis distance. It turns out that in many practical applications the solution that minimizes the Sampson error coincides with the exact ML solution up to several significant digits; usually, two or three rounds of Sampson error modification are sufficient[11),14),15)].

However, minimizing the Sampson error is not straightforward. Many numerical schemes have been proposed, including the *FNS* (*Fundamental Numerical Scheme*) of Chojnacki et al.[4)], the *HEIV* (*Heteroscedastic Errors-in-Variable*) of Leedan and Meer[19)] and Matei and Meer[20)], and the *projective Gauss-Newton iterations* of Kanatani and Sugaya[13)]. All these rely on local search, but the iterations do not always converge if not started from a value sufficiently close to the solution. Hence, accurate approximation schemes that do not require iterations are very much desired, even though the solution may not be optimal, and various algebraic methods have been studied in the past.

## 5. Algebraic Fitting

For the sake of brevity, we abbreviate $\boldsymbol{\xi}^{(k)}(\boldsymbol{x}_\alpha)$ as $\boldsymbol{\xi}_\alpha^{(k)}$. *Algebraic fitting* refers to minimizing the *algebraic distance*

$$J = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L}(\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{\theta})^2 = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L}\boldsymbol{\theta}^\top\boldsymbol{\xi}_\alpha^{(k)}\boldsymbol{\xi}_\alpha^{(k)\top}\boldsymbol{\theta} = (\boldsymbol{\theta}, \boldsymbol{M}\boldsymbol{\theta}), \tag{13}$$

where we define

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L}\boldsymbol{\xi}_\alpha^{(k)}\boldsymbol{\xi}_\alpha^{(k)\top}. \tag{14}$$

Equation (13) is trivially minimized by $\boldsymbol{\theta} = \boldsymbol{0}$ unless some scale normalization is imposed on $\boldsymbol{\theta}$. The most common normalization is $\|\boldsymbol{\theta}\| = 1$, which we call the *standard LS*. However, the solution depends on the normalization. So, we naturally ask: *What normalization will maximize the accuracy of the solution?* This question was raised first by Al-Sharadqah and Chernov[1)] and Rangarajan and Kanatani[23)] for circle fitting, then by Kanatani and Rangarajan[12)] for ellipse fitting and by Niitsuma et al.[22)] for homography computation. In this paper, we generalize these results to an arbitrary number of constraints. Following these authors[1),12),22),23)], we consider the class of normalizations

$$(\boldsymbol{\theta}, \boldsymbol{N}\boldsymbol{\theta}) = c, \tag{15}$$

with some symmetric matrix $\boldsymbol{N}$ for a nonzero constant $c$. In Eq. (15), $\boldsymbol{\theta}$ is the optimization parameter and $\boldsymbol{N}$ is an unknown matrix to be determined, while the constant $c$ is fixed for the problem. We need not specify the value of $c$, because $\boldsymbol{N}$ is unknown. Since Eq. (15) can be written as $(\boldsymbol{\theta}, (\boldsymbol{N}/c)\boldsymbol{\theta}) = 1$, we may determine $\boldsymbol{N}' = \boldsymbol{N}/c$ instead of $\boldsymbol{N}$, but the form of Eq. (15) with $c$ unspecified is more convenient in our analysis.

Traditionally, the matrix $\boldsymbol{N}$ is positive definite or semidefinite, but in the following, we allow $\boldsymbol{N}$ to be nondefinite (i.e., neither positive nor negative definite), so the constant $c$ in Eq. (15) is not necessarily positive. The standard treatment of algebraic fitting goes as follows. Given the matrix $\boldsymbol{N}$, the solution $\boldsymbol{\theta}$ that minimizes Eq. (13) subject to Eq. (15), if it exists, is given by the solution of the generalized eigenvalue problem

$$\boldsymbol{M}\boldsymbol{\theta} = \lambda\boldsymbol{N}\boldsymbol{\theta}. \tag{16}$$

Note that $\boldsymbol{M}$ is always positive semidefinite from its definition in Eq. (14). If there is no noise in the data, we have $(\boldsymbol{\theta}, \boldsymbol{\xi}_\alpha^{(k)}) = 0$ for all $k$ and $\alpha$. Hence, Eq. (14) implies $\boldsymbol{M}\boldsymbol{\theta} = \boldsymbol{0}$, so $\lambda = 0$. In the presence of noise, $\boldsymbol{M}$ is positive definite, so $\lambda$ is positive whether $\boldsymbol{N}$ is positive definite or semidefinite. The corresponding solution is obtained as the eigenvector $\boldsymbol{\theta}$ for the smallest $\lambda$. For the standard LS, for which $\boldsymbol{N} = \boldsymbol{I}$, Eq. (16) becomes an ordinary eigenvalue problem

$$\boldsymbol{M}\boldsymbol{\theta} = \lambda\boldsymbol{\theta}, \tag{17}$$

and the solution is the unit eigenvector $\boldsymbol{\theta}$ of $\boldsymbol{M}$ for the smallest eigenvalue $\lambda$.

This is the traditional treatment of algebraic fitting, but the situation is slightly different here: $\boldsymbol{N}$ is not yet given and can be nondefinite, and the eigenvalues of Eq. (16) may not be all positive. So, we face the problem of which eigenvalues and eigenvectors of Eq. (16) to choose as a solution. In the following, we do perturbation analysis[10] of Eq. (16) by assuming that $\lambda \approx 0$ and choose the solution to be the eigenvector $\boldsymbol{\theta}$ for the $\lambda$ with the smallest absolute value, although in theory there remains a possibility that another choice happens to produce a better result in some cases. We also regard Eq. (16) as the definition of "algebraic fitting," rather than Eq. (13) and Eq. (15). This is because, while Eq. (16) always has a solution, Eq. (13) may not be minimized subject to Eq. (15) by a finite $\boldsymbol{\theta}$. This can occur, for example, when the contour of $(\boldsymbol{\theta}, \boldsymbol{M}\boldsymbol{\theta})$, which is a hyperellipsoid in the space of $\boldsymbol{\theta}$, happens to be elongated in a direction in the null space of $\boldsymbol{N}$. Then, the minimum of $(\boldsymbol{\theta}, \boldsymbol{M}\boldsymbol{\theta})$ could be reached in the limit of $\|\boldsymbol{\theta}\| \to \infty$. Note that Eq. (15) is unable to normalize the norm $\|\boldsymbol{\theta}\|$ into a finite value if $\boldsymbol{N}$ has a null space. Theoretically, such an anomaly can always occur because $\boldsymbol{M}$ is a random variable defined by noisy data, and if the probability of such an occurrence is nearly 0, it may still lead to $E[\|\hat{\boldsymbol{\theta}}\|] = \infty$[2]. Once the problem is converted to Eq. (16), for which eigenvectors $\boldsymbol{\theta}$ have scale indeterminacy, we can adopt normalization $\|\boldsymbol{\theta}\| = 1$ rather than Eq. (13). Then, the solution $\boldsymbol{\theta}$ is always a unit vector.

## 6. Error Analysis

We can expand each $\boldsymbol{\xi}_\alpha^{(k)}$ in the form

$$\boldsymbol{\xi}_\alpha^{(k)} = \bar{\boldsymbol{\xi}}_\alpha^{(k)} + \Delta_1\boldsymbol{\xi}_\alpha^{(k)} + \Delta_2\boldsymbol{\xi}_\alpha^{(k)} + \cdots, \tag{18}$$

where $\bar{\boldsymbol{\xi}}_\alpha^{(k)}$ is the noiseless value, and $\Delta_i\boldsymbol{\xi}_\alpha^{(k)}$ is the $i$th order term in $\Delta\boldsymbol{x}_\alpha$. The first order term is written as

$$\Delta_1\boldsymbol{\xi}_\alpha^{(k)} = \boldsymbol{T}_\alpha^{(k)}\Delta\boldsymbol{x}_\alpha, \qquad \boldsymbol{T}_\alpha^{(k)} \equiv \left.\frac{\partial\boldsymbol{\xi}^{(k)}(\boldsymbol{x})}{\partial\boldsymbol{x}}\right|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_\alpha}. \tag{19}$$

We define the covariance matrices of $\boldsymbol{\xi}_\alpha^{(k)}$, $k = 1, \ldots, L$, by

$$V^{(kl)}[\boldsymbol{\xi}_\alpha] \equiv E[\Delta_1\boldsymbol{\xi}_\alpha^{(k)}\Delta_1\boldsymbol{\xi}_\alpha^{(l)\top}] = \boldsymbol{T}_\alpha^{(k)}E[\Delta\boldsymbol{x}_\alpha\Delta\boldsymbol{x}_\alpha^\top]\boldsymbol{T}_\alpha^{(l)\top} = \boldsymbol{T}_\alpha^{(k)}V[\boldsymbol{x}_\alpha]\boldsymbol{T}_\alpha^{(l)\top}, \tag{20}$$

where $E[\,\cdot\,]$ denotes expectation.

The *Sampson error* that we mentioned in Section 4, which approximates the minimum of the Mahalanobis distance in Eq. (11) subject to the constraints in Eq. (12), has the following form[7,8]:

$$K(\boldsymbol{\theta}) = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l,=1}^{L} W_\alpha^{(kl)}(\boldsymbol{\xi}_\alpha^{(k)}, \boldsymbol{\theta})(\boldsymbol{\xi}_\alpha^{(l)}, \boldsymbol{\theta}). \tag{21}$$

Here, $W_\alpha^{(kl)}$ is the $(kl)$ element of $(\boldsymbol{V}_\alpha)_r^-$, and $\boldsymbol{V}_\alpha$ is the matrix whose $(kl)$ element is

$$\boldsymbol{V}_\alpha = \Big((\boldsymbol{\theta}, V^{(kl)}[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})\Big), \tag{22}$$

where the true data values $\bar{\boldsymbol{x}}_\alpha$ in the definition of $V^{(kl)}[\boldsymbol{\xi}_\alpha]$ are replaced by their observations $\boldsymbol{x}_\alpha$. The operation $(\,\cdot\,)_r^-$ denotes the pseudoinverse of truncated rank $r$ (i.e., with all eigenvalues except the largest $r$ replaced by 0 in the spectral decomposition), and $r$ is the rank (the number of independent equations) of the constraint in Eq. (12). The name "Sampson error" stems from the classical ellipse fitting scheme of Sampson[25]. For given $\boldsymbol{x}_\alpha^{(k)}$, Eq. (21) can be minimized by various means including the FNS[4], HEIV[19,20], and the projective Gauss-Newton iteration[13].

**Example 4 (Ellipse fitting).** For the ellipse fitting in Example 1, the first order error $\Delta_1\boldsymbol{\xi}$ is written as

$$\Delta_1\boldsymbol{\xi}_\alpha = 2\begin{pmatrix} \bar{x}_\alpha & \bar{y}_\alpha & 0 & 1 & 0 & 0 \\ 0 & \bar{x}_\alpha & \bar{y}_\alpha & 0 & 1 & 0 \end{pmatrix}^\top \begin{pmatrix} \Delta x_\alpha \\ \Delta y_\alpha \end{pmatrix}. \tag{23}$$

The second order error $\Delta_2\boldsymbol{\xi}_\alpha$ has the following form:

$$\Delta_2\boldsymbol{\xi}_\alpha = (\Delta x_\alpha^2, 2\Delta x_\alpha\Delta y_\alpha, \Delta y_\alpha^2, 0, 0, 0)^\top. \tag{24}$$

**Example 5 (Fundamental matrix computation).** For the fundamental matrix computation in Example 2, the first order error $\Delta_1\boldsymbol{\xi}$ is written as

$$\Delta_1\boldsymbol{\xi}_\alpha = \begin{pmatrix} \bar{x}'_\alpha & \bar{y}'_\alpha & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{x}'_\alpha & \bar{y}'_\alpha & 1 & 0 & 0 & 0 \\ \bar{x}_\alpha & 0 & 0 & \bar{y}_\alpha & 0 & 0 & 1 & 0 & 0 \\ 0 & \bar{x}_\alpha & 0 & 0 & \bar{y}_\alpha & 0 & 0 & 1 & 0 \end{pmatrix}^\top \begin{pmatrix} \Delta x_\alpha \\ \Delta y_\alpha \\ \Delta x'_\alpha \\ \Delta y'_\alpha \end{pmatrix}. \tag{25}$$

The second order error $\Delta_2\boldsymbol{\xi}_\alpha$ has the following form:

$$\Delta_2\boldsymbol{\xi}_\alpha = (\Delta x_\alpha \Delta x'_\alpha, \Delta x_\alpha \Delta y'_\alpha, 0, \Delta y_\alpha \Delta x'_\alpha, \Delta y_\alpha \Delta y'_\alpha, 0, 0, 0, 0)^\top. \tag{26}$$

**Example 6 (Homography computation).** For the fundamental matrix computation in Example 2, the first order error $\Delta_1\boldsymbol{\xi}$ is written as

$$\Delta_1\boldsymbol{\xi}^{(1)}_\alpha = \begin{pmatrix} 0 & 0 & 0 & -1 & 0 & 0 & \bar{y}'_\alpha & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & \bar{y}'_\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \bar{x}_\alpha & \bar{y}_\alpha & 1 \end{pmatrix}^\top \begin{pmatrix} \Delta x_\alpha \\ \Delta y_\alpha \\ \Delta x'_\alpha \\ \Delta y'_\alpha \end{pmatrix},$$

$$\Delta_1\boldsymbol{\xi}^{(2)}_\alpha = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -\bar{x}'_\alpha & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -\bar{x}'_\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\bar{x}_\alpha & -\bar{y}_\alpha & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^\top \begin{pmatrix} \Delta x_\alpha \\ \Delta y_\alpha \\ \Delta x'_\alpha \\ \Delta y'_\alpha \end{pmatrix},$$

$$\Delta_1\boldsymbol{\xi}^{(3)}_\alpha = \begin{pmatrix} -\bar{y}'_\alpha & 0 & 0 & \bar{x}'_\alpha & 0 & 0 & 0 & 0 & 0 \\ 0 & -\bar{y}'_\alpha & 0 & 0 & \bar{x}'_\alpha & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \bar{x}_\alpha & \bar{y}_\alpha & 1 & 0 & 0 & 0 \\ -\bar{x}_\alpha & -\bar{y}_\alpha & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^\top \begin{pmatrix} \Delta x_\alpha \\ \Delta y_\alpha \\ \Delta x'_\alpha \\ \Delta y'_\alpha \end{pmatrix}. \tag{27}$$

The second order error $\Delta_2\boldsymbol{\xi}^{(k)}_\alpha$ has the following form:

$$\begin{aligned} \Delta_2\boldsymbol{\xi}^{(1)}_\alpha &= (0,0,0,0,0,0,\Delta x_\alpha \Delta y'_\alpha, \Delta y_\alpha \Delta y'_\alpha, 0)^\top, \\ \Delta_2\boldsymbol{\xi}^{(2)}_\alpha &= (0,0,0,0,0,0,-\Delta x'_\alpha \Delta x_\alpha, -\Delta x'_\alpha \Delta y_\alpha, 0)^\top, \\ \Delta_2\boldsymbol{\xi}^{(3)}_\alpha &= (-\Delta y'_\alpha \Delta x_\alpha, -\Delta y'_\alpha \Delta y_\alpha, 0, \Delta x'_\alpha \Delta x_\alpha, \Delta x'_\alpha \Delta y_\alpha, 0,0,0,0)^\top. \end{aligned} \tag{28}$$

## 7. Perturbation Analysis

Substituting Eq. (18) into Eq. (14), we obtain

$$\boldsymbol{M} = \bar{\boldsymbol{M}} + \Delta_1\boldsymbol{M} + \Delta_2\boldsymbol{M} + \cdots, \tag{29}$$

where

$$\bar{\boldsymbol{M}} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L} \bar{\boldsymbol{\xi}}^{(k)}_\alpha \bar{\boldsymbol{\xi}}^{(k)\top}_\alpha, \tag{30}$$

$$\Delta_1\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L} (\bar{\boldsymbol{\xi}}^{(k)}_\alpha \Delta_1\boldsymbol{\xi}^{(k)\top}_\alpha + \Delta_1\boldsymbol{\xi}^{(k)}_\alpha \bar{\boldsymbol{\xi}}^{(k)\top}_\alpha), \tag{31}$$

$$\Delta_2\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L} (\bar{\boldsymbol{\xi}}^{(k)}_\alpha \Delta_2\boldsymbol{\xi}^{(k)\top}_\alpha + \Delta_1\boldsymbol{\xi}^{(k)}_\alpha \Delta_1\boldsymbol{\xi}^{(k)\top}_\alpha + \Delta_2\boldsymbol{\xi}^{(k)}_\alpha \bar{\boldsymbol{\xi}}^{(k)\top}_\alpha). \tag{32}$$

We also expand the solution $\boldsymbol{\theta}$ and $\lambda$ of Eq. (16) in the form

$$\boldsymbol{\theta} = \bar{\boldsymbol{\theta}} + \Delta_1\boldsymbol{\theta} + \Delta_2\boldsymbol{\theta} + \cdots, \qquad \lambda = \bar{\lambda} + \Delta_1\lambda + \Delta_2\lambda + \cdots. \tag{33}$$

Substituting Eq. (29) and Eq. (33) into Eq. (16), we have

$$\begin{aligned} &(\bar{\boldsymbol{M}} + \Delta_1\boldsymbol{M} + \Delta_2\boldsymbol{M} + \cdots)(\bar{\boldsymbol{\theta}} + \Delta_1\boldsymbol{\theta} + \Delta_2\boldsymbol{\theta} + \cdots) \\ &= (\bar{\lambda} + \Delta_1\lambda + \Delta_2\lambda + \cdots)\boldsymbol{N}(\bar{\boldsymbol{\theta}} + \Delta_1\boldsymbol{\theta} + \Delta_2\boldsymbol{\theta} + \cdots). \end{aligned} \tag{34}$$

Note that $\boldsymbol{N}$ is a variable to be determined, not a given function of observations, so it is not expanded. Since we consider perturbations near the true values, the resulting matrix $\boldsymbol{N}$ may be a function of the true data values. In that event, we replace the true data values by their observations and do an a posteriori analysis to see how this affects the accuracy. For the moment, we regard $\boldsymbol{N}$ as an unknown variable. From a strictly mathematical point of view, the two sides of Eq. (34) may not define an absolutely convergent series expansion. Here, we do not go into such a theoretical question; we simply test the usefulness of the final results by experiments a posteriori, as commonly done in physics and engineering. At any rate, we are concerned with only up to the second order terms in the subsequent analysis.

Equating terms of the same order in Eq. (34), we obtain

$$\bar{\boldsymbol{M}}\bar{\boldsymbol{\theta}} = \bar{\lambda}\boldsymbol{N}\bar{\boldsymbol{\theta}}, \tag{35}$$

$$\bar{\boldsymbol{M}}\Delta_1\boldsymbol{\theta} + \Delta_1\boldsymbol{M}\bar{\boldsymbol{\theta}} = \bar{\lambda}\boldsymbol{N}\Delta_1\boldsymbol{\theta} + \Delta_1\lambda\boldsymbol{N}\bar{\boldsymbol{\theta}}, \tag{36}$$

$$\bar{\boldsymbol{M}}\Delta_2\boldsymbol{\theta} + \Delta_1\boldsymbol{M}\Delta_1\boldsymbol{\theta} + \Delta_2\boldsymbol{M}\bar{\boldsymbol{\theta}} = \bar{\lambda}\boldsymbol{N}\Delta_2\boldsymbol{\theta} + \Delta_1\lambda\boldsymbol{N}\Delta_1\boldsymbol{\theta} + \Delta_2\lambda\boldsymbol{N}\bar{\boldsymbol{\theta}}. \tag{37}$$

We have $\bar{\boldsymbol{M}}\bar{\boldsymbol{\theta}} = \boldsymbol{0}$ for the true values, so $\bar{\lambda} = 0$. From Eq. (31), we have $(\bar{\boldsymbol{\theta}}, \Delta_1\bar{\boldsymbol{M}}\bar{\boldsymbol{\theta}}) = 0$. Computing the inner product of Eq. (36) and $\bar{\boldsymbol{\theta}}$ on both sides, we see that $\Delta_1\lambda = 0$. Multiplying Eq. (36) by the pseudoinverse $\bar{\boldsymbol{M}}^-$ of $\bar{\boldsymbol{M}}$ from left, we obtain

$$\Delta_1\boldsymbol{\theta} = -\bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}\bar{\boldsymbol{\theta}}. \tag{38}$$

Note that since $\bar{\boldsymbol{M}}\bar{\boldsymbol{\theta}} = \boldsymbol{0}$, the matrix $\bar{\boldsymbol{M}}^{-}\bar{\boldsymbol{M}}$ ($\equiv \boldsymbol{P}_{\bar{\boldsymbol{\theta}}}$) is the projection operator in the direction orthogonal to $\bar{\boldsymbol{\theta}}$. Also, equating the first order terms in the expansion $\|\bar{\boldsymbol{\theta}} + \Delta_1\boldsymbol{\theta} + \Delta_2\boldsymbol{\theta} + \cdots\|^2 = 1$ shows $(\bar{\boldsymbol{\theta}}, \Delta_1\boldsymbol{\theta}) = 0$[10], hence $\boldsymbol{P}_{\bar{\boldsymbol{\theta}}}\Delta_1\boldsymbol{\theta} = \Delta_1\boldsymbol{\theta}$. Substituting Eq. (38) into Eq. (37) and computing its inner product with $\bar{\boldsymbol{\theta}}$ on both sides, we obtain

$$\Delta_2\lambda = \frac{(\bar{\boldsymbol{\theta}}, \Delta_2\boldsymbol{M}\bar{\boldsymbol{\theta}}) - (\bar{\boldsymbol{\theta}}, \Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \boldsymbol{N}\bar{\boldsymbol{\theta}})} = \frac{(\bar{\boldsymbol{\theta}}, \boldsymbol{T}\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \boldsymbol{N}\bar{\boldsymbol{\theta}})}, \tag{39}$$

where we put

$$\boldsymbol{T} = \Delta_2\boldsymbol{M} - \Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}. \tag{40}$$

Next, we consider the second order error $\Delta_2\boldsymbol{\theta}$. Since $\boldsymbol{\theta}$ is normalized to unit norm, we are interested in the error component orthogonal to $\bar{\boldsymbol{\theta}}$. So, we consider

$$\Delta_2^{\perp}\boldsymbol{\theta} \equiv \boldsymbol{P}_{\bar{\boldsymbol{\theta}}}\Delta_2\boldsymbol{\theta} \; (= \bar{\boldsymbol{M}}^{-}\bar{\boldsymbol{M}}\Delta_2\boldsymbol{\theta}). \tag{41}$$

Multiplying Eq. (37) by $\bar{\boldsymbol{M}}^{-}$ from left and substituting Eq. (38), we obtain

$$\begin{aligned}\Delta_2^{\perp}\boldsymbol{\theta} &= \Delta_2\lambda\bar{\boldsymbol{M}}^{-}\boldsymbol{N}\bar{\boldsymbol{\theta}} + \bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{M}}^{-}\Delta_2\boldsymbol{M}\bar{\boldsymbol{\theta}} \\ &= \frac{(\bar{\boldsymbol{\theta}}, \boldsymbol{T}\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \boldsymbol{N}\bar{\boldsymbol{\theta}})}\bar{\boldsymbol{M}}^{-}\boldsymbol{N}\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{M}}^{-}\boldsymbol{T}\bar{\boldsymbol{\theta}}. \end{aligned} \tag{42}$$

## 8. Covariance and Bias

### 8.1 Covariance Analysis

From Eq. (38), the covariance matrix $V[\boldsymbol{\theta}]$ of the solution $\boldsymbol{\theta}$ has the leading term

$$\begin{aligned}V[\boldsymbol{\theta}] &= E[\Delta_1\boldsymbol{\theta}\Delta_1\boldsymbol{\theta}^{\top}] = \frac{1}{N^2}\bar{\boldsymbol{M}}^{-}E[(\Delta_1\boldsymbol{M}\boldsymbol{\theta})(\Delta_1\boldsymbol{M}\boldsymbol{\theta})^{\top}]\bar{\boldsymbol{M}}^{-} \\ &= \frac{1}{N^2}\bar{\boldsymbol{M}}^{-}E\Big[\sum_{\alpha=1}^{N}\sum_{k=1}^{L}(\Delta\boldsymbol{\xi}_{\alpha}^{(k)}, \boldsymbol{\theta})\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}\sum_{\beta=1}^{N}\sum_{l=1}^{L}(\Delta\boldsymbol{\xi}_{\beta}^{(l)}, \boldsymbol{\theta})\bar{\boldsymbol{\xi}}_{\beta}^{(l)\top}\Big]\bar{\boldsymbol{M}}^{-} \\ &= \frac{1}{N^2}\bar{\boldsymbol{M}}^{-}\sum_{\alpha,\beta=1}^{N}\sum_{k,l=1}^{L}(\boldsymbol{\theta}, E[\Delta\boldsymbol{\xi}_{\alpha}^{(k)}\Delta\boldsymbol{\xi}_{\beta}^{(l)\top}]\boldsymbol{\theta})\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}\bar{\boldsymbol{\xi}}_{\beta}^{(l)\top}\bar{\boldsymbol{M}}^{-} \\ &= \frac{1}{N^2}\bar{\boldsymbol{M}}^{-}\Big(\sum_{\alpha=1}^{N}\sum_{k,l=1}^{L}(\boldsymbol{\theta}, V^{(kl)}[\boldsymbol{\xi}_{\alpha}]\boldsymbol{\theta})\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}\bar{\boldsymbol{\xi}}_{\alpha}^{(l)\top}\Big)\bar{\boldsymbol{M}}^{-} \\ &= \frac{1}{N}\bar{\boldsymbol{M}}^{-}\bar{\boldsymbol{M}}'\bar{\boldsymbol{M}}^{-}, \end{aligned} \tag{43}$$

where we define

$$\bar{\boldsymbol{M}}' = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{L}(\boldsymbol{\theta}, V^{(kl)}[\boldsymbol{\xi}_{\alpha}]\boldsymbol{\theta})\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}\bar{\boldsymbol{\xi}}_{\alpha}^{(l)\top} \tag{44}$$

In the above derivation, we have noted that from our noise assumption we have $E[\Delta_1\boldsymbol{\xi}_{\alpha}^{(k)}\Delta_1\boldsymbol{\xi}_{\beta}^{(l)\top}] = \delta_{\alpha\beta}V^{(kl)}[\boldsymbol{\xi}_{\alpha}]$, where $\delta_{\alpha\beta}$ is the Kronecker delta.

### 8.2 Bias Analysis

The important observation is that the covariance matrix $V[\boldsymbol{\theta}]$ does not contain $\boldsymbol{N}$. Thus, *all algebraic methods have the same covariance matrix in the leading order*, as pointed out by Al-Sharadqah and Chernov[1] for circle fitting. This observation leads us to focus on the bias. We now seek an $\boldsymbol{N}$ that reduces the bias as much as possible. It would be desirable if we could find such an $\boldsymbol{N}$ that minimizes the total mean square error $E[\|\Delta_1\boldsymbol{\theta}+\Delta_2\boldsymbol{\theta}+\cdots\|^2]$, but at the moment this seems to be an intractable problem; minimizing the bias alone is a practical compromise, whose effectiveness is tested by experiments a posteriori.

From Eq. (38), we see that the first order bias $E[\Delta_1\boldsymbol{\theta}]$ is $\boldsymbol{0}$, hence the leading bias is $E[\Delta_2^{\perp}\boldsymbol{\theta}]$. From Eq. (42), we have

$$E[\Delta_2^{\perp}\boldsymbol{\theta}] = \frac{(\bar{\boldsymbol{\theta}}, E[\boldsymbol{T}]\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \boldsymbol{N}\bar{\boldsymbol{\theta}})}\bar{\boldsymbol{M}}^{-}\boldsymbol{N}\bar{\boldsymbol{\theta}} - \bar{\boldsymbol{M}}^{-}E[\boldsymbol{T}]\bar{\boldsymbol{\theta}}. \tag{45}$$

We now evaluate the expectation $E[\boldsymbol{T}]$ of $\boldsymbol{T}$ in Eq. (40). From Eq. (32), we see that $E[\Delta_2\boldsymbol{M}]$ is given by

$$\begin{aligned}E[\Delta_2\boldsymbol{M}] &= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L}\Big(\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}E[\Delta_2\boldsymbol{\xi}_{\alpha}^{(k)}]^{\top} + E[\Delta_1\boldsymbol{\xi}_{\alpha}^{(k)}\Delta_1\boldsymbol{\xi}_{\alpha}^{(k)\top}] + E[\Delta_2\boldsymbol{\xi}_{\alpha}^{(k)}]\bar{\boldsymbol{\xi}}_{\alpha}^{(k)\top}\Big) \\ &= \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L}\Big(V^{(kk)}[\boldsymbol{\xi}_{\alpha}] + 2\mathcal{S}[\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}\boldsymbol{e}_{\alpha}^{(k)\top}]\Big), \end{aligned} \tag{46}$$

where we have used Eq. (20) and defined

$$\boldsymbol{e}_\alpha^{(k)} \equiv E[\Delta_2 \boldsymbol{\xi}_\alpha^{(k)}]. \tag{47}$$

The operator $\mathcal{S}[\,\cdot\,]$ denotes symmetrization ($\mathcal{S}[\boldsymbol{A}] = (\boldsymbol{A}+\boldsymbol{A}^\top)/2$). The expectation $E[\Delta_1 \boldsymbol{M}\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M}]$ has the following form (see Appendix):

$$E[\Delta_1 \boldsymbol{M}\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M}] = \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{L}\Big(\mathrm{tr}[\bar{\boldsymbol{M}}^- V^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} \\ +(\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V^{(kl)}[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[V^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big). \tag{48}$$

From Eq. (46) and Eq. (48), the expectation of $\boldsymbol{T}$ is

$$E[\boldsymbol{T}] = \boldsymbol{N}_\mathrm{T} - \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{L}\Big(\mathrm{tr}[\bar{\boldsymbol{M}}^- V^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} \\ +(\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V^{(kl)}[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[V^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big), \tag{49}$$

where we put

$$\boldsymbol{N}_\mathrm{T} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k=1}^{L}\Big(V^{(kk)}[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[\bar{\boldsymbol{\xi}}_\alpha^{(k)}\boldsymbol{e}_\alpha^{(k)\top}]\Big). \tag{50}$$

## 9. HyperLS

Now, let us choose $\boldsymbol{N}$ to be the expression $E[\boldsymbol{T}]$ in Eq. (49) itself, namely,

$$\boldsymbol{N} = \boldsymbol{N}_\mathrm{T} - \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{L}\Big(\mathrm{tr}[\bar{\boldsymbol{M}}^- V^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} + (\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(l)})V^{(kl)}[\boldsymbol{\xi}_\alpha] \\ +2\mathcal{S}[V^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big), \tag{51}$$

Letting $\boldsymbol{N} = E[\boldsymbol{T}]$ in Eq. (45), we see that

$$E[\Delta_2^\perp \boldsymbol{\theta}] = \bar{\boldsymbol{M}}^-\Big(\frac{(\bar{\boldsymbol{\theta}}, \boldsymbol{N}\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \boldsymbol{N}\bar{\boldsymbol{\theta}})}\boldsymbol{N} - \boldsymbol{N}\Big)\bar{\boldsymbol{\theta}} = \boldsymbol{0}. \tag{52}$$

Since the right-hand side of Eq. (49) contains the true values $\bar{\boldsymbol{\xi}}_\alpha$ and $\bar{\boldsymbol{M}}$, we replace $\bar{\boldsymbol{x}}_\alpha$ in their definition by the observation $\boldsymbol{x}_\alpha$. This does not affect the result, since the odd order noise terms have expectation 0 and hence the resulting error in $E[\Delta_2^\perp \boldsymbol{\theta}]$ is of the fourth order. Thus, the second order bias is *exactly* 0.

In fitting a circle to a point sequence, Al-Sharadqah and Chernov[1] proposed to choose $\boldsymbol{N} = \boldsymbol{N}_\mathrm{T}$ and showed that the second order bias $E[\Delta_2^\perp \boldsymbol{\theta}]$ is zero *up to* $O(1/N^2)$. They called their method *Hyper*. What we have shown here is that the second order bias is *completely* removed by including the second term on the right-hand side of Eq. (51). We call our scheme *HyperLS*.

Note that $\boldsymbol{N}$ has scale indeterminacy: If $\boldsymbol{N}$ is multiplied by $c$ ($\neq 0$), Eq. (16) has the same solution $\boldsymbol{\theta}$; only $\lambda$ is divided by $c$. Thus, the noise characteristics $V^{(kl)}[\boldsymbol{\xi}_\alpha]$ in Eq. (20) and hence $V[\boldsymbol{x}_\alpha]$ need to be known only up to scale; *we need not know the absolute magnitude of the noise*.

For numerical computation, standard linear algebra routines for solving the generalized eigenvalue problem of Eq. (16) assume that $\boldsymbol{N}$ is positive definite, but here $\boldsymbol{N}$ is nondefinite. This causes no problem, because Eq. (16) can be written as

$$\boldsymbol{N}\boldsymbol{\theta} = \frac{1}{\lambda}\boldsymbol{M}\boldsymbol{\theta}. \tag{53}$$

As mentioned earlier, the matrix $\boldsymbol{M}$ in Eq. (14) is positive definite for noisy data, so we can solve Eq. (53) instead, using a standard routine. If the smallest eigenvalue of $\boldsymbol{M}$ happens to be 0, it indicates that the data are all exact, so any method, e.g., the standard LS, gives an exact solution. For noisy data, the solution $\boldsymbol{\theta}$ is given by the eigenvector of Eq. (53) for the eigenvalue $1/\lambda$ with the largest absolute value.

**Example 7 (Ellipse fitting).** If the noise in $(x_\alpha, y_\alpha)$ is independent and Gaussian with mean 0 and standard deviation $\sigma$, the vector $\boldsymbol{e}_\alpha$ ($= \boldsymbol{e}^{(1)}$) in Eq. (47) is given by

$$\boldsymbol{e}_\alpha = \sigma^2(1, 0, 1, 0, 0, 0)^\top. \tag{54}$$

Hence, the matrix $\boldsymbol{N}_\mathrm{T}$ in Eq. (50) is given by

$$\boldsymbol{N}_{\rm T} = \frac{\sigma^2}{N}\sum_{\alpha=1}^{N}\begin{pmatrix} 6x_\alpha^2 & 6x_\alpha y_\alpha & x_\alpha^2+y_\alpha^2 & 6x_\alpha & 2y_\alpha & 1 \\ 6x_\alpha y_\alpha & 4(x_\alpha^2+y_\alpha^2) & 6x_\alpha y_\alpha & 4y_\alpha & 4x_\alpha & 0 \\ x_\alpha^2+y_\alpha^2 & 6x_\alpha y_\alpha & 6y_\alpha^2 & 2x_\alpha & 6y_\alpha & 1 \\ 6x_\alpha & 4y_\alpha & 2x_\alpha & 4 & 0 & 0 \\ 2y_\alpha & 4x_\alpha & 6y_\alpha & 0 & 4 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \qquad (55)$$

The Taubin method[26)] is to use as $\boldsymbol{N}$

$$\boldsymbol{N}_{\rm Taubin} = \frac{4\sigma^2}{N}\sum_{\alpha=1}^{N}\begin{pmatrix} x_\alpha^2 & x_\alpha y_\alpha & 0 & x_\alpha & 0 & 0 \\ x_\alpha y_\alpha & x_\alpha^2+y_\alpha^2 & x_\alpha y_\alpha & y_\alpha & x_\alpha & 0 \\ 0 & x_\alpha y_\alpha & y_\alpha^2 & 0 & y_\alpha & 0 \\ x_\alpha & y_\alpha & 0 & 1 & 0 & 0 \\ 0 & x_\alpha & y_\alpha & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \qquad (56)$$

which we see is obtained by letting $\boldsymbol{e}_\alpha = \boldsymbol{0}$ in Eq. (50). As pointed out earlier, the value of $\sigma$ in Eq. (55) and Eq. (56) need not be known. Hence, we can simply let $\sigma = 1$ in Eq. (16) and Eq. (53) in actual computation.

**Example 8 (Fundamental matrix computation).** If the noise in $(x_\alpha, y_\alpha)$ and $(x'_\alpha, y'_\alpha)$ is independent and Gaussian with mean 0 and standard deviation $\sigma$, the vector $\boldsymbol{e}_\alpha$ $(= \boldsymbol{e}^{(1)})$ in Eq. (47) is $\boldsymbol{0}$, so the $\boldsymbol{N}_{\rm T}$ in Eq. (50) becomes

$$\boldsymbol{N}_{\rm T} = \frac{\sigma^2}{N}\sum_{\alpha=1}^{N}\begin{pmatrix} x_\alpha^2+x_\alpha'^2 & x'_\alpha y'_\alpha & x'_\alpha & x_\alpha y_\alpha & 0 & 0 & x_\alpha & 0 & 0 \\ x'_\alpha y'_\alpha & x_\alpha^2+y_\alpha'^2 & y'_\alpha & 0 & x_\alpha y_\alpha & 0 & 0 & x_\alpha & 0 \\ x'_\alpha & y'_\alpha & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_\alpha y_\alpha & 0 & 0 & y_\alpha^2+x_\alpha'^2 & x'_\alpha y'_\alpha & x'_\alpha & y_\alpha & 0 & 0 \\ 0 & x_\alpha y_\alpha & 0 & x'_\alpha y'_\alpha & y_\alpha^2+y_\alpha'^2 & y'_\alpha & 0 & y_\alpha & 0 \\ 0 & 0 & 0 & x'_\alpha & y'_\alpha & 1 & 0 & 0 & 0 \\ x_\alpha & 0 & 0 & y_\alpha & 0 & 0 & 1 & 0 & 0 \\ 0 & x_\alpha & 0 & 0 & y_\alpha & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \qquad (57)$$

It turns out that the use of this matrix $\boldsymbol{N}_{\rm T}$ coincides with the well known Taubin method[26)]. As in ellipse fitting, we can let $\sigma = 1$ in Eq. (57) in actual computation.

**Example 9 (Homography computation).** If the noise in $(x_\alpha, y_\alpha)$ and $(x'_\alpha, y'_\alpha)$ is independent and Gaussian with mean 0 and standard deviation $\sigma$, the vectors $\boldsymbol{e}_\alpha^{(k)}$ in Eq. (47) are all $\boldsymbol{0}$, so the $\boldsymbol{N}_{\rm T}$ in Eq. (50) becomes

$$\boldsymbol{N}_{\rm T} = \frac{\sigma^2}{N}\sum_{\alpha=1}^{N}\left(\begin{matrix} x_\alpha^2+y_\alpha'^2+1 & x_\alpha y_\alpha & x_\alpha & -x'_\alpha y'_\alpha & 0 \\ x_\alpha y_\alpha & y_\alpha^2+y_\alpha'^2+1 & y_\alpha & 0 & -x'_\alpha y'_\alpha \\ x_\alpha & y_\alpha & 1 & 0 & 0 \\ -x'_\alpha y'_\alpha & 0 & 0 & x_\alpha^2+x_\alpha'^2+1 & x_\alpha y_\alpha \\ 0 & -x'_\alpha y'_\alpha & 0 & x_\alpha y_\alpha & y_\alpha^2+x_\alpha'^2+1 \\ 0 & 0 & 0 & x_\alpha & y_\alpha \\ -x'_\alpha & 0 & 0 & -y'_\alpha & 0 \\ 0 & -x'_\alpha & 0 & 0 & -x'_\alpha \\ 0 & 0 & 0 & 0 & 0 \end{matrix}\right.$$

$$\left.\begin{matrix} 0 & -x'_\alpha & 0 & 0 \\ 0 & 0 & -x'_\alpha & 0 \\ 0 & 0 & 0 & 0 \\ x_\alpha & -y'_\alpha & 0 & 0 \\ y_\alpha & 0 & -y'_\alpha & 0 \\ 1 & 0 & 0 & 0 \\ 0 & x_\alpha^2+x_\alpha'^2+y_\alpha'^2 & 2x_\alpha y_\alpha & 2x_\alpha \\ 0 & 2x_\alpha y_\alpha & y_\alpha^2+x_\alpha'^2+y_\alpha'^2 & 2y_\alpha \\ 0 & 2x_\alpha & 2y_\alpha & 2 \end{matrix}\right), \qquad (58)$$

For homography computation, the constraint is a vector equation in Eq. (8). Hence, the Taubin method[26)], which is defined for a single constraint equation, cannot be applied. However, the use of the above $\boldsymbol{N}_{\rm T}$ as $\boldsymbol{N}$ plays the same role of the Taubin method[26)] for ellipse fitting and fundamental matrix computation, as first pointed out by Rangarajan and Papamichalis[24)]. As before, we can let $\sigma$ = 1 in the matrix $\boldsymbol{N}_{\rm T}$ in actual computation.

In the following, we call the use of $\boldsymbol{N}_{\rm T}$ as $\boldsymbol{N}$ the *Taubin approximation*. For fundamental matrix computation, it coincides with the Taubin method[26], but for homography computation the Taubin method was not defined. For ellipse fitting, the Taubin method and the Taubin approximation are slightly different; the Taubin method is equivalent to use only the first term on the right hand side of Eq. (50). For circle fitting, the Taubin approximation is the same as the "Hyper" of Al-Sharadqah and Chernov[1].

## 10. Numerical Experiments

We did the following three experiments:

**Ellipse fitting:** We fit an ellipse to the point sequence shown in **Fig. 2**(a). We took 31 equidistant points on the first quadrant of an ellipse with major and minor axes 100 and 50 pixels, respectively.

**Fundamental matrix computation:** We compute the fundamental matrix between the two images shown in Fig. 2(b), which view a cylindrical grid surface from two directions. The image size is assumed to be 600 × 600 (pixels) with focal lengths 600 pixels for both. The 91 grid points are used as corresponding points.

**Homography computation:** We compute the homography relating the two images shown in Fig. 2(c), which view a planar grid surface from two directions. The image size is assumed to be 800 × 800 (pixels) with focal lengths 600 pixels for both. The 45 grid points are used as corresponding points.

In all experiments, we divided the data coordinate values by 600 (pixels) (i.e., we used 600 pixels as the unit of length) to make all the data values within the range of about ±1. This is for stabilizing numerical computation with finite precision length; without this data scale normalization, serious accuracy loss is incurred, as pointed out by Hartley[5] for fundamental matrix computation.

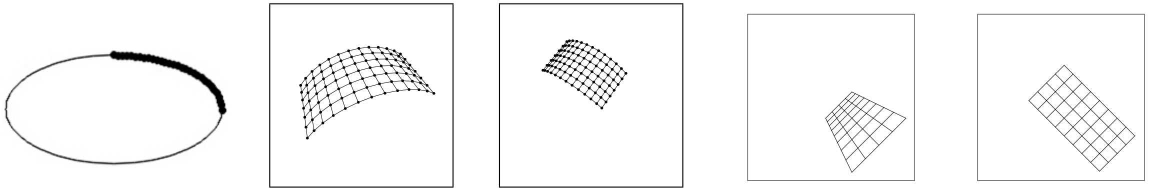
For each example, we compared the standard LS, HyperLS, its Taubin approximation, and ML, for which we used the FNS of Chojnacki et al.[4] for ellipse fitting and fundamental matrix computation and the multiconstraint FNS of Niitsuma et al.[22] for homography computation. As mentioned in Section 4, FNS minimizes not directly Eq. (11) but the Sampson error in Eq. (21), and the exact ML solution can be obtained by repeated Sampson error minimization[16]. The the solution that minimizes the Sampson error usually agrees with the ML solution up to several significant digits[11,14,15]. Hence, FNS can safely be regarded as minimizing Eq. (11).

Let $\bar{\boldsymbol{\theta}}$ be the true value of the parameter $\boldsymbol{\theta}$, and $\hat{\boldsymbol{\theta}}$ its computed value. We consider the following error:

$$\Delta^{\perp}\boldsymbol{\theta} = \boldsymbol{P}_{\bar{\boldsymbol{\theta}}}\hat{\boldsymbol{\theta}}, \qquad \boldsymbol{P}_{\bar{\boldsymbol{\theta}}} \equiv \boldsymbol{I} - \bar{\boldsymbol{\theta}}\bar{\boldsymbol{\theta}}^{\top}. \tag{59}$$

The matrix $\boldsymbol{P}_{\bar{\boldsymbol{\theta}}}$ represents the orthogonal projection onto the space orthogonal to $\bar{\boldsymbol{\theta}}$. Since the computed value $\hat{\boldsymbol{\theta}}$ is normalized to a unit vector, it distributes around $\bar{\boldsymbol{\theta}}$ on the unit sphere. Hence, the meaningful deviation is its component orthogonal to $\bar{\boldsymbol{\theta}}$, so we measure the error component in the tangent space to the unit sphere at $\bar{\boldsymbol{\theta}}$ (**Fig. 3**).

We added independent Gaussian noise of mean 0 and standard deviation $\sigma$ to the $x$ and $y$ coordinates of data each point and repeated the fitting $M$ times for each $\sigma$, using different noise. We let $M = 10000$ for ellipse fitting and fundamental matrix computation and $M = 1000$ for homography computation. Then, we



(a) (b) (c)

**Fig. 2** (a) 31 points on an ellipse. (b) Two views of a curved grid. (c) Two views of a planar grid.

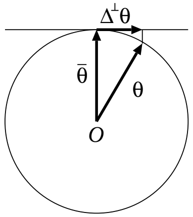
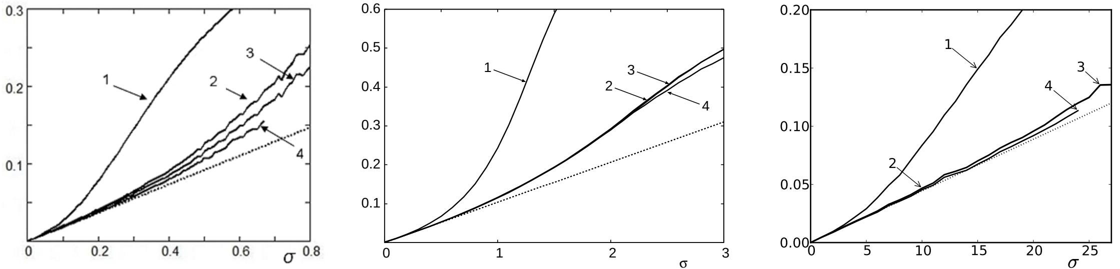


**Fig. 3** The true value $\bar{\boldsymbol{\theta}}$, the computed value $\boldsymbol{\theta}$, and its orthogonal component $\Delta\boldsymbol{\theta}$ to $\bar{\boldsymbol{\theta}}$.



(a) (b) (c)

**Fig. 4** RMS error vs. the standard deviation $\sigma$ of the noise added to each point: 1. standard LS, 2. Taubin approximation, 3. HyperLS, 4. ML. The dotted lines indicate the KCR lower bound. (a) Ellipse fitting. (b) Fundamental matrix computation. (c) Homography computation.

evaluated the root-mean-square (RMS) error

$$E = \sqrt{\frac{1}{M}\sum_{a=1}^{M} \|\Delta^{\perp}\boldsymbol{\theta}^{(a)}\|^2}, \tag{60}$$

where $\Delta\boldsymbol{\theta}^{(a)}$ is the value of $\Delta\boldsymbol{\theta}$ in the $a$th trial. The theoretical accuracy limit, called the *KCR lower bound*3),8),10), is given by

$$E[\Delta^{\perp}\boldsymbol{\theta}\Delta^{\perp}\boldsymbol{\theta}^{\top}] \succ \frac{\sigma^2}{N}\Big(\frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{L} \bar{W}_{\alpha}^{(kl)}\bar{\boldsymbol{\xi}}_{\alpha}^{(k)}\bar{\boldsymbol{\xi}}_{\alpha}^{(l)\top}\Big)^{-} \equiv V_{\rm KCR}[\boldsymbol{\theta}], \tag{61}$$

where $\bar{W}_{\alpha}^{(kl)}$ is the value of $W_{\alpha}^{(kl)}$ in Eq. (21) evaluated by assuming $\sigma = 1$ and using the true values $\bar{\boldsymbol{\theta}}$ and $\bar{\boldsymbol{\xi}}_{\alpha}^{(kl)}$. The relation $\succ$ means that the left-hand side minus the right-hand side is a positive semidefinite symmetric matrix, and the operation $(\,\cdot\,)^{-}$ denotes pseudoinverse. We compared the RMS error in Eq. (60) with the trace Eq. (61):

$$\sqrt{E[\|\Delta^{\perp}\boldsymbol{\theta}\|^2]} \geq \sqrt{{\rm tr}V_{\rm KCR}[\boldsymbol{\theta}]}. \tag{62}$$

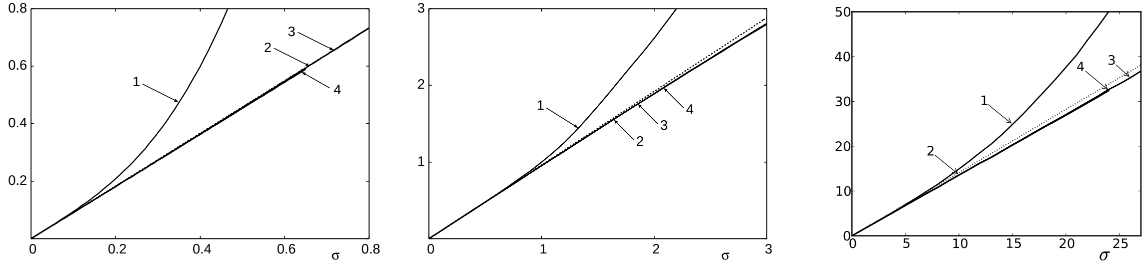
**Figure 4** plots for $\sigma$ the RMS error of Eq. (60) for each method and the KCR lower bound of Eq. (62).

We also compared the reprojection error for different methods. According to statistics, the reprojection error $I$ in Eq. (11) for ML is subject to a $\chi^2$ distribution with $rN - d$ degrees of freedom, where $r$ is the codimension of the constraint and $d$ is the dimension of the parameters8). Hence, if ML is computed by assuming $\sigma = 1$, the square root of the average reprojection error per datum is expected to be $\sigma\sqrt{r - d/N}$. **Figure 5** plots the square root of the average, per datum, of the computed reprojection error, which was approximated by the Sampson error $K(\boldsymbol{\theta})$ in Eq. (21), along with the theoretical expectation.

We observe the following:

**Ellipse fitting:** The standard LS performs poorly, while ML exhibits the highest accuracy, almost reaching the KCR lower bound. However, ML computation fails to converge above a certain noise level. In contrast, HyperLS produces, without iterations, an accurate solution close to ML. The accuracy of its Taubin approximation is practically the same as the traditional Taubin method and is slightly lower than HyperLS. Since $r = 1$, $d = 5$, the square root of the average reprojection error per datum has theoretical expectation $\sigma\sqrt{1 - 5/N}$ to a first approximation. We see that the computed value almost coincides with the expected value expect for the standard LS.

**Fundamental matrix computation:** Again, the standard LS is poor, while ML has the highest accuracy, almost reaching the KCR lower bound. The accuracy of HyperLS is very close to ML. Its Taubin approximation (= the traditional Taubin method) has practically the same accuracy as HyperLS.



(a) (b) (c)

**Fig. 5** Root square average reprojection error per datum vs. the standard deviation $\sigma$ of the noise added to each point: 1. standard LS, 2. Taubin approximation, 3. HyperLS, 4. ML. The dotted lines indicate theoretical expectation. (a) Ellipse fitting. (b) Fundamental matrix computation. (c) Homography computation.

The fundamental matrix has the constraint that its rank be 2. The comparison here is done before the rank constraint is imposed. Hence, $r = 1$, $d = 8$, and the square root of the average reprojection error per datum is expected to be $\sigma\sqrt{1 - 8/N}$. We see that the computed value almost coincides with the expected value expect for the standard LS.

**Homography computation:** In this case, too, the standard LS is poor, while ML has the highest accuracy, almost reaching the KCR lower bound. However, ML computation fails to converge above a certain noise level. The accuracy of HyperLS is very close to ML. Its Taubin approximation has practically the same accuracy as HyperLS. Since $r = 2$, $d = 4$, the square root of the average reprojection error per datum is expected to be $\sigma\sqrt{2(1 - 4/N)}$. We see that the computed value almost coincides with the expected value expect for the standard LS.

In all examples, the standard LS performs poorly, while ML provides the highest accuracy. Note that the differences among different methods are more marked when measured in the RMS error than in the reprojection error. This is because the RMS error measures the error of the parameters of the equation, while the reprojection error measures the closeness of the fit to the data. For ellipse fitting, for example, the RMS error compares the fitted ellipse equation and the true ellipse equation, while the reprojection error measures how close the fitted ellipse is to the data. As a result, even if two ellipse have nearly the same distances to the data, their shapes can be very different. This difference becomes more conspicuous as the data cover a shorter segment of the ellipse. The same observation can be done for fundamental matrix computation and homography computation.

We also see from our experiments that ML computation may fail in the presence of large noise. The convergence of ML critically depends on the accuracy of the initialization. In the above experiments, we used the standard LS to start the FNS iterations. We confirmed that the use of HyperLS to start the iterations significantly extends the noise range of convergence, though the computation fails sooner or later. On the other hand, HyperLS is algebraic and hence immune to the convergence problem, producing a solution close in accuracy to ML in any noise level.

The Taubin approximation is clearly inferior to HyperLS for ellipse fitting but is almost equivalent to HyperLS for fundamental matrices and homographies. This reflects the fact that while $\boldsymbol{\xi}$ is *quadratic* in $x$ and $y$ for ellipses (see Eq. (4)), the corresponding $\boldsymbol{\xi}$ and $\boldsymbol{\xi}^{(k)}$ are *bilinear* in $x$, $y$, $x'$, and $y'$ for fundamental matrices (see Eq. (6)) and homographies (see Eq. (9)), so $\boldsymbol{e}_\alpha^{(k)}$ in Eq. (47) is $\mathbf{0}$. In structure-from-motion applications, we frequently do inference from multiple images based on "multilinear" constraints involving homographies, fundamental matrices, trifocal tensors, and other geometric quantities[7)]. For such problems, the constraint itself is nonlinear but is *linear in observations of each image*.

Then, $\boldsymbol{e}_\alpha^{(k)} = \mathbf{0}$, because noise in different images are assumed to be independent. In such a problem, the accuracy of HyperLS is nearly the same as its Taubin approximation. However, HyperLS is clearly superior in a situation where the constraint involves nonlinear terms in observations of the same image, e.g., ellipse fitting.

## 11. Concluding Remarks

We have presented a general formulation for a special type of least squares (LS) estimator, which we call "HyperLS," for geometric problems that frequently appear in vision applications. We described the problem in the most general terms and discussed various theoretical issues that have not been fully studied so far. In particular, we pointed out that the characteristics of image-based inference is very different to the conventional statistical domains and discussed in detail various issues related to ML and algebraic fitting. Then, we derived HyperLS by introducing a normalization that eliminates statistical bias of LS up to second order noise terms.

It would be ideal if we could minimize the total mean squares error by taking all higher order terms into account. Due to technical difficulties, we limited our attention to the bias up to the second order. Also, we introduced in our derivation several assumptions about the choice of the eigenvalues and the convergence of series expansion. However, the purpose of this paper is not to establish mathematical theorems with formal proofs. Our aim is to derive techniques that are useful in practical problems; the usefulness is to be tested by experiments.

Our numerical experiments for computing ellipses, fundamental matrices, and homographies showed that HyperLS yields a solution far superior to the standard LS and comparable in accuracy to ML, which is known to produce highly accurate solutions but may fail to converge if poorly initialized. Thus, HyperLS is a perfect candidate for ML initialization. We compared the performance of HyperLS and its Taubin approximation and attributed the performance differences to the structure of the problem. In this paper, we did not show real image demos, concentrating on the general mathematical framework, because particular applications have been shown elsewhere[1),12),22),23)].

**Acknowledgments** The authors thank Ali Al-Sharadqah and Nikolai Chernov of the University of Alabama at Birmingham, U.S.A, Wolfgang Förstner of the University of Bonn, Germany, and Alexander Kukush of National Taras Shevchenko University of Kyiv, Ukraine, for helpful discussions. 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) Al-Sharadqah, A. and Chernov, N.: Error analysis for circle fitting algorithms. *Elec. J. Stat.*, Vol.3, pp.886–911 (2009).
2) Cheng, C.-L. and Kukush, A.: Non-existence of the first moment of the adjusted least squares estimator in multivariate errors-in-variables model, *Metrika*, Vol.64, No.1, pp.41–46 (2006).
3) Chernov, N. and Lesort, C.: Statistical efficiency of curve fitting algorithms, *Comput. Stat. Data Anal.*, Vol.47, Vol.4, pp.713–728 (2004).
4) 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).
5) Hartley, R.I.: In defense of the eight-point algorithm, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.19, No.6, pp.580–593 (1997).
6) Hartley, R. and Kahl, F.: Optimal algorithms in multiview geometry, *Proc. 8th Asian Conf. Computer Vision*, Tokyo, Japan, November 2007, Vol.1, pp.13–34 (2007).
7) Hartley, R. and Zisserman, A.: *Multiple View Geometry in Computer Vision*, 2nd ed., Cambridge University Press, Cambridge, U.K. (2004).
8) Kanatani, K.: *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier Science, Amsterdam, The Netherlands (1996); reprinted, Dover, New York, U.S.A. (2005).
9) Kanatani, K.: Ellipse fitting with hyperaccuracy, *IEICE Trans. Inf. & Syst.*, Vol.E89-D, No.10, pp.2653–2660 (2006).
10) Kanatani, K.: Statistical optimization for geometric fitting: Theoretical accuracy analysis and high order error analysis, *Int. J. Comput. Vis.*, Vol.80, No.2, pp.167–188 (2006).
11) Kanatani, K. and Niitsuma, H.: Optimal two-view planar scene triangulation, *IPSJ Trans. Comput. Vis. Appl.*, Vol.4 (2011), to appear.
12) Kanatani, K. and Rangarajan, P.: Hyper least squares fitting of circles and ellipses, *Comput. Stat. Data Anal.*, Vol.55, Vol.6, pp.2197–2208 (2011).
13) Kanatani, K. and Sugaya, Y.: Performance evaluation of iterative geometric fitting algorithms, *Comp. Stat. Data Anal.*, Vol.52, No.2, pp.1208–1222 (2007).

14) Kanatani, K. and Sugaya, Y.: Compact algorithm for strictly ML ellipse fitting, *Proc. 19th Int. Conf. Pattern Recog.*, Tampa, FL, U.S.A. (2008).

15) Kanatani, K. and Sugaya, Y.: Compact fundamental matrix computation, *IPSJ Trans. Comput. Vis. Appl.*, Vol.2, pp.59–70 (2010).

16) Kanatani, K. and Sugaya, Y.: Unified computation of strict maximum likelihood for geometric fitting, *J. Math. Imaging Vis.*, Vol.38, pp.1–13 (2010).

17) Kukush, K., Markovski, I. and Van Huffel, S.: Consistent fundamental matrix estimation in a quadratic measurement error model arising in motion analysis, *Comp. Stat. Data Anal.*, Vol.41, No.1, pp.3–18 (2002).

18) Kukush, K., Markovski, I. and Van Huffel, S.: Consistent estimation in an implicit quadratic measurement error model, *Comp. Stat. Data Anal.*, Vol.47, No.1, pp.123–147 (2004).

19) Leedan, Y. and Meer, P.: Heteroscedastic regression in computer vision: Problems with bilinear constraint, *Int. J. Comput. Vis.*, Vol.37, No.2, pp.127–150 (2000).

20) Matei, B.C. and Meer, P.: Estimation of nonlinear errors-in-variables models for computer vision applications, *IEEE Trans. Patt. Anal. Mach. Intell.*, Vol.28, No.10, pp.1537–1552 (2006).

21) Neyman, J. and Scott, E.L.: Consistent estimates based on partially consistent observations, *Econometrica*, Vol.16, No.1, pp.1–32 (1948).

22) Niitsuma, H., Rangarajan, P. and Kanatani, K.: High accuracy homography computation without iterations, *Proc. 16th Symp. Sensing Imaging Inf.*, Yokohama, Japan (2010).

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

24) Rangarajan, P. and Papamichalis, P.: Estimating homographies without normalization, *Proc. Int. Conf. Image Process.*, Cairo, Egypt, pp.3517–3520 (2009).

25) Sampson, P.D.: Fitting conic sections to "very scattered" data: An iterative refinement of the Bookstein algorithm, *Comput. Graphics Image Process.*, Vol.18, No.1, pp.97–108 (1982).

26) Taubin, G.: 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.*, Vol.13, No.11, pp.1115–1138 (1991).

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

## Appendix

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\Big[\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}}^-\frac{1}{N}\sum_{\beta=1}^{N}\sum_{l=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\beta^{(l)}\Delta_1\boldsymbol{\xi}_\beta^{(l)\top}\\
&\quad +\Delta_1\boldsymbol{\xi}_\beta^{(l)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}\Big)\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}}^-(\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} + \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} + \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}+\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} + \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}+(\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} + \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} + (\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} + (\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{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}}^-\delta_{\alpha\beta}V^{(kl)}[\boldsymbol{\xi}_\alpha] + \mathrm{tr}[\bar{\boldsymbol{M}}^-\delta_{\alpha\beta}V^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\beta^{(l)\top}
\end{aligned}
$$

$$
\begin{aligned}
&+(\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\beta^{(l)})\delta_{\alpha\beta} V^{(kl)}[\boldsymbol{\xi}_\alpha] + \delta_{\alpha\beta} V^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha^{(k)} \bar{\boldsymbol{\xi}}_\beta^{(l)\top}\Big) \\
&= \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big(\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}\bar{\boldsymbol{M}}^- V^{(kl)}[\boldsymbol{\xi}_\alpha] + \mathrm{tr}[\bar{\boldsymbol{M}}^- V^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} \\
&\quad +(\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha^{(l)}) V^{(kl)}[\boldsymbol{\xi}_\alpha] + V^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}\Big) \\
&= \frac{1}{N^2}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3}\Big(\mathrm{tr}[\bar{\boldsymbol{M}}^- V^{(kl)}[\boldsymbol{\xi}_\alpha]]\bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top} + (\bar{\boldsymbol{\xi}}_\alpha^{(k)}, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha^{(l)}) V^{(kl)}[\boldsymbol{\xi}_\alpha] \\
&\quad +2\mathcal{S}[V^{(kl)}[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha^{(k)}\bar{\boldsymbol{\xi}}_\alpha^{(l)\top}]\Big). \qquad (63)
\end{aligned}
$$

Thus, Eq. (48) is obtained.

(Received January 1, 2011)
(Revised May 16, 2011)
(Accepted August 1, 2011)
(Released October 17, 2011)

(Communicated by *Peter Sturm*)

**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.

**Prasanna Rangarajan** received his B.E. in electronics and communication engineering from Bangalore University, Bangalore, India, in 2000 and his M.S. in electrical engineering from Columbia University, New York, NY, U.S.A., in 2003. He is currently a Ph.D. candidate in electrical engineering at Southern Methodist University, Dallas, TX, U.S.A. His research interests include image processing, structured illumination and parameter estimation for computer vision.

**Yasuyuki Sugaya** received his B.E., M.S., and Ph.D. in computer science from the University of Tsukuba, Ibaraki, Japan, in 1996, 1998, and 2001, respectively. From 2001 to 2006, he was Assistant Professor of computer science at Okayama University, Okayama, Japan. Currently, he is Associate Professor of information and computer sciences at Toyohashi University of Technology, Toyohashi, Aichi, Japan. His research interests include image processing and computer vision. He received the IEICE best paper award in 2005.

**Hirotaka Niitsuma** received his B.E. and M.S. in applied physics from Osaka University, Japan, in 1993 and 1995, respectively, and his Ph.D. in informatic science 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.