Memoirs of the Faculty of Engineering, Okayama University, Vol. 47, pp. 1–18, January 2013

Overviews of Optimization Techniques for Geometric Estimation

Kenichi KANATANI*
Department of Computer Science, Okayama University
Okayama 700-8530 Japan

(Received November 26, 2012)

We summarize techniques for optimal geometric estimation from noisy observations for computer vision applications. We first discuss the interpretation of optimality and point out that geometric estimation is different from the standard statistical estimation. We also describe our noise modeling and a theoretical accuracy limit called the KCR lower bound. Then, we formulate estimation techniques based on minimization of a given cost function: least squares (LS), maximum likelihood (ML), which includes reprojection error minimization as a special case, and Sampson error minimization. We describe bundle adjustment and the FNS scheme for numerically solving them and the hyperaccurate correction that improves the accuracy of ML. Next, we formulate estimation techniques not based on minimization of any cost function: iterative reweight, renormalization, and hyper-renormalization. Finally, we show numerical examples to demonstrate that hyper-renormalization has higher accuracy than ML, which has widely been regarded as the most accurate method of all. We conclude that hyper-renormalization is robust to noise and currently is the best method.

Contents

*E-mail kanatani@suri.cs.okayama-u.ac.jp

1. Introduction

One of the most fundamental tasks of computer vision is to compute the 2-D and 3-D shapes of objects based on *geometric constraints*, by which we mean properties of the scene that can be described by relatively simple equations such as the objects being lines or planes, their being parallel or orthogonal, and the camera imaging geometry being perspective projection. We call the inference based on such geometric constraints *geometric estimation*. If there is no noise in the observation, this is a simple calculus and poses no technical problems. In the presence of noise, however, the assumed constraints do not exactly hold. For "optimal" geometric estimation in the presence of noise, a lot of efforts have been made since 1980s by many researchers including the author. This paper summarizes that history and reports the latest results.

2. Preliminaries

2.1 Optimization of geometric estimation

"Optimization" of geometric estimation has a slightly different meaning from the conventional optimization, which means computing a solution that maximizes or minimizes a given cost function. Finding a solution that maximizes the profit, the gain, or the efficiency or minimizes the loss, the error, or the delay is the most fundamental problem in all domains of engineering. The aim of geometric estimation in computer vision, on the other hand, is simply to obtain a solution to a given equation. However, the difficulty lies in the fact that *the equation has no solution if it is defined using noisy observations*. Hence, the task is: assuming that the equation has a unique solution in the absence of noise, we *estimate* it. We call the solution that would be obtained in the absence of noise the *true value*. In this sense, estimation of the true value critically relies on the properties of the noise. Thus, the task of geometric estimation that we consider in this paper is:

> Appropriately assuming the statistical properties of the noise, we infer the solution of equations defined in terms of noisy observations.

We could solve this problem by reducing it to the conventional optimization of minimizing a cost function, but we need not necessarily do so. This is the main message of this paper.

2.2 Definition of geometric estimation

The geometric estimation problem we consider here is defined as follows. We observe some quantity $\boldsymbol{x}$ (a vector), which is assumed to satisfy in the absence of noise an equation

$$F(\boldsymbol{x};\boldsymbol{\theta}) = 0, \tag{1}$$

parameterized by unknown vector $\boldsymbol{\theta}$. This equation is called the *geometric constraint*. Our task is to estimate the parameter $\boldsymbol{\theta}$ from noisy instances $\boldsymbol{x}_\alpha$, $\alpha = 1, ..., N$, of $\boldsymbol{x}$. Many computer vision problems are formulated in this way, and we can compute from the estimated $\boldsymbol{\theta}$ the positions, the shapes, and the motions of the objects we are viewing. In many problems, we can reparameterize the problem so that the constraint is liner in the parameter $\boldsymbol{\theta}$ (but generally nonlinear in the data $\boldsymbol{x}$), and Eq. (1) has the form

$$(\boldsymbol{\xi}(\boldsymbol{x}), \boldsymbol{\theta}) = 0, \tag{2}$$

where $\boldsymbol{\xi}(\boldsymbol{x})$ is a vector-valued nonlinear mapping of $\boldsymbol{x}$. In this paper, we denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $(\boldsymbol{a}, \boldsymbol{b})$. Equation (2) implies that the scale of $\boldsymbol{\theta}$ is indeterminate, so we hereafter normalize $\boldsymbol{\theta}$ to unit norm: $\|\boldsymbol{\theta}\| = 1$.

Example 1 (line fitting). To a given point sequence (x_α, y_α), $\alpha = 1, ..., N$, we fit a line

$$Ax + By + C = 0. \tag{3}$$

(Fig. 1(a)). If we define

$$\boldsymbol{\xi}(x, y) \equiv (x,\, y,\, 1)^\top, \qquad \boldsymbol{\theta} \equiv (A,\, B,\, C)^\top, \tag{4}$$

Eq. (3) is written as

$$(\boldsymbol{\xi}(x, y), \boldsymbol{\theta}) = 0. \tag{5}$$

Example 2 (ellipse fitting) To a given point sequence (x_α, y_α), $\alpha = 1, ..., N$, we fit an ellipse

$$Ax^2 + 2Bxy + Cy^2 + 2(Dx + Ey) + F = 0. \tag{6}$$

(Fig. 1(b)). If we define

$$\begin{aligned} \boldsymbol{\xi}(x, y) &\equiv (x^2,\, 2xy,\, y^2,\, 2x,\, 2y,\, 1)^\top, \\ \boldsymbol{\theta} &\equiv (A,\, B,\, C,\, D,\, E,\, F)^\top, \end{aligned} \tag{7}$$

Eq. (6) is written as

$$(\boldsymbol{\xi}(x, y), \boldsymbol{\theta}) = 0. \tag{8}$$

Example 3 (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* (Fig. 1(c))

$$\left(\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \boldsymbol{F} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix}\right) = 0, \tag{9}$$

where $\boldsymbol{F}$ is called the *fundamental matrix*, from which we can compute the camera positions and the 3-D structure of the scene [12]. If we define

$$\begin{aligned} &\boldsymbol{\xi}(x, y, x', y') \equiv (xx',\, xy',\, x,\, yx',\, yy',\, y,\, x',\, y',\, 1)^\top, \\ &\boldsymbol{\theta} \equiv (F_{11},\, F_{12},\, F_{13},\, F_{21},\, F_{22},\, F_{23},\, F_{31},\, F_{32},\, F_{33})^\top, \end{aligned} \tag{10}$$

Eq. (9) is written as

$$(\boldsymbol{\xi}(x, y, x', y'), \boldsymbol{\theta}) = 0. \tag{11}$$

In these examples, the constraint is a single equation (a scalar equation), but the following arguments can be straightforwardly extended to multiple equations (a vector equation). However, we would need a lot of indices, which makes the description rather complicated. For simplicity, we focus only on the single constraint case in this paper.

Note that the vectors in Eqs. (4), (7), and (10) contain the constant 1 in their components. If the input values (x, y) and (x', y') are very large, the constant 1 may be ignored in the course of finite length numerical computation, resulting in considerable accuracy

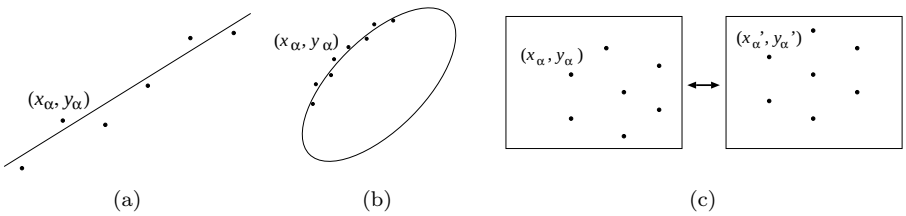


Figure 1: (a) Line fitting. (b) Ellipse fitting. (c) Fundamental matrix computation.

loss. To prevent this, we need to divide the input values by an appropriate reference length so that x, y, x', and y' are all $O(1)$ [10]. To concentrate on theories, however, we do not consider for the moment such techniques for finite length numerical computation.

We have so far assumed that the true values $\bar{\boldsymbol{x}}_\alpha$ of the observations $\boldsymbol{x}_\alpha$ exactly satisfy Eq. (2). In reality, however, different types of values that do not satisfy Eq. (2) even in the absence noise often mix in the data due to imperfection of image processing operations. Such values are called *outliers*, while the values that should satisfy the constraint in the absence of noise are called *inliers*. Because we cannot assume any properties about outliers except that they do not satisfy the constraint, theoretical analysis of outliers is very difficult. One of the basic treatments for detecting outliers is *voting*: we compute the parameter $\boldsymbol{\theta}$ by assuming that the data are inliers and iteratively test if the result agrees with all the data or a part of the data. Popular voting techniques include *RANSAC* (*Random Sampling Consensus*) [8] and *LMedS* (*Least Median of Squares*) [40]. Estimation that is not very much affected by outliers is called *robust estimation*, to which various *M-estimators* [13], which alleviate the influence of those data largely deviated from the constraint, are introduced. Since any outlier detection technique must be coupled with estimation for inliers, we consider in this paper only inlier data.

2.3 Modeling of noise

In the context of image analysis, "noise" means *uncertainty of image processing operations*, rather than random fluctuations over time or space as commonly understood in physics and communications. This reflects the fact that standard image processing operations such as feature extraction and edge detection are not perfect and do not necessarily output exactly what we are looking for. We model this data uncertainty in statistical terms: the observed value $\boldsymbol{x}_\alpha$ is regarded as a perturbation from its true value $\bar{\boldsymbol{x}}_\alpha$ by an independent random Gaussian variable $\Delta\boldsymbol{x}_\alpha$ of mean $\boldsymbol{0}$ and covariance matrix $V[\boldsymbol{x}_\alpha]$. Furthermore, the covariance matrix $V[\boldsymbol{x}_\alpha]$ is assumed to be known *up to scale*. Namely, we express it in the form

$$V[\boldsymbol{x}_\alpha] = \sigma^2 V_0[\boldsymbol{x}_\alpha] \tag{12}$$

for some unknown constant σ, which we call the *noise level*. The matrix $V_0[\boldsymbol{x}_\alpha]$, which we call the *normalized covariance matrix*, describes the orientation dependence of the uncertainty in relative terms and is assumed to be known. The separation of $V[\boldsymbol{x}_\alpha]$ into σ^2 and $V_0[\boldsymbol{x}_\alpha]$ is merely a matter of convenience; there is no fixed rule. This convention is motivated by the fact that estimation of the absolute magnitude of data uncertainty is very difficult in practice, while optimal estimation can be done, as shown shortly, only from the knowledge of $V_0[\boldsymbol{x}_\alpha]$.

If the observation $\boldsymbol{x}_\alpha$ is regarded as a random variable in this way, its nonlinear mapping $\boldsymbol{\xi}(\boldsymbol{x}_\alpha)$, which we write $\boldsymbol{\xi}_\alpha$ for short, is also a random variable. Its covariance matrix $V[\boldsymbol{\xi}_\alpha] = \sigma^2 V_0[\boldsymbol{\xi}_\alpha]$ is evaluated to a first approximation in terms of the Jacobi matrix $\partial\boldsymbol{\xi}/\partial\boldsymbol{x}$ of the mapping $\boldsymbol{\xi}(\boldsymbol{x})$ as follows:

$$V_0[\boldsymbol{\xi}_\alpha] = \left.\frac{\partial\boldsymbol{\xi}}{\partial\boldsymbol{x}}\right|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_\alpha} V_0[\boldsymbol{x}_\alpha] \left.\frac{\partial\boldsymbol{\xi}}{\partial\boldsymbol{x}}\right|^\top_{\boldsymbol{x}=\bar{\boldsymbol{x}}_\alpha}. \tag{13}$$

This expression contains the true value $\bar{\boldsymbol{x}}_\alpha$, which in actual computation is replaced by the observation $\boldsymbol{x}_\alpha$. It has been confirmed by experiments that this replacement does not practically affect the final result. It has also been confirmed that upgrading the first approximation to higher orders does not have any practical effect.

Strictly speaking, if the noise in $\boldsymbol{x}_\alpha$ is Gaussian, the noise in its nonlinear mapping $\boldsymbol{\xi}_\alpha$ is no longer Gaussian, although it is expected to be Gaussian-like when the noise is small. The effect of approximating this Gaussian-like distribution by a Gaussian distribution is discussed later.

2.4 Statistical models

What we call "geometric estimation" has much in common with the standard estimation problems treated in most textbooks of statistics, which we simply call "statistical estimation", but many different features exist. The standard statistical estimation is formulated as follows. Given observations $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_N$, we regard them as independent samples from a probability density $p(\boldsymbol{x}|\boldsymbol{\theta})$ parameterized by an unknown $\boldsymbol{\theta}$, and we estimate $\boldsymbol{\theta}$ from $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_N$. The probability density $p(\boldsymbol{x}|\boldsymbol{\theta})$ is called the *statistical model*, which explains the underlying mechanism of the phe-

nomenon we are observing. In other words, the purpose of statistical estimation is to understand the hidden source of our observations. Naturally, we understand it better as the number of observations increases. In this sense, accuracy vs. the number N of observations in the asymptotic limit $N \to \infty$ is a major concern of statistics. Basically, there exist two approaches for statistical estimation:

Minimization principle

We seek the value $\boldsymbol{\theta}$ that minimizes a specified cost function $J(\boldsymbol{x}_1, ..., \boldsymbol{x}_N; \boldsymbol{\theta})$. The most widely used approach is *maximum likelihood* (*ML*), which minimizes

$$J = -\sum_{\alpha=1}^{N} \log p(\boldsymbol{x}_\alpha|\boldsymbol{\theta}). \tag{14}$$

The motivation is to maximize the *likelihood* $\prod_{\alpha=1}^{N} p(\boldsymbol{x}_\alpha|\boldsymbol{\theta})$, but for the convenience of computation, its negative logarithm is minimized. We can further introduce the a priori probability $p(\boldsymbol{\theta})$ of the parameter $\boldsymbol{\theta}$ and minimize

$$J = -\sum_{\alpha=1}^{N} \log p(\boldsymbol{x}_\alpha|\boldsymbol{\theta}) - \log p(\boldsymbol{\theta}). \tag{15}$$

This is equivalent to maximizing the *a posteriori probability* obtained by the Bayes theorem and called the *maximum a posteriori probability* (*MAP*). This is a special case of *Bayesian estimation*, which minimizes the *Bayes risk* defined in terms of the estimated a posteriori probability density function.

Method of estimating functions

We compute the value $\boldsymbol{\theta}$ that satisfies a set of equations in the form

$$\boldsymbol{g}(\boldsymbol{x}_1, ..., \boldsymbol{x}_N; \boldsymbol{\theta}) = \boldsymbol{0}, \tag{16}$$

which is called the *estimating equation* [9]. The function $\boldsymbol{g}$ is called the *estimating function*. If we choose $\boldsymbol{g}$ to be

$$\boldsymbol{g} = -\sum_{\alpha=1}^{N} \nabla_{\boldsymbol{\theta}} \log p(\boldsymbol{x}_\alpha|\boldsymbol{\theta}), \tag{17}$$

the problem reduces to ML, where $\nabla_{\boldsymbol{\theta}}$ denotes the (vector-valued) derivative with respect to $\boldsymbol{\theta}$. Thus, the estimating equations can be regarded as generalization of the minimization principle. However, the estimating function $\boldsymbol{g}$ need not be derivatives of any cost function; we can define or modify $\boldsymbol{g}$ in such a way that the solution $\boldsymbol{\theta}$ has desirable properties. The desirable properties include *unbiasedness*, *consistency*, and *efficiency*. In this sense, the method of estimating functions is more general and more flexible with a possibility of yielding better solutions than the minimization principle.

2.5 Geometric models

One of the most prominent distinctions of the geometric estimation problems we are considering from the standard statistical estimation problems is that the starting point is Eq. (1), which merely states that the true values should satisfy this implicit equation. We call Eq. (1) (or Eq. (2)) the *geometric model*, which only specifies the necessary constraint and does not explain the mechanism as to how the data $\boldsymbol{x}_\alpha$ are generated. Hence, we cannot express $\boldsymbol{x}_\alpha$ in terms of the parameter $\boldsymbol{\theta}$ as an explicit function.

Another big difference is that while statistical estimation is based on *repeated* observations regarded as samples from the statistical model (= probability density), geometric estimation is done from *one* set of data $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_N\}$ *for that problem*. Naturally, the estimation accuracy increases with less noise (= less observation uncertainty). In this sense, accuracy vs. the noise level σ in the limit $\sigma \to 0$ is a major concern [19].

In computer vision applications, the asymptotic analysis for $N \to \infty$ does not have much sense, because the number of data obtained by image processing operations is limited in number. Usually, the output of an image processing operation is accompanied by its reliability index, and we select only those data that have high reliability indices. If we want to increase the number of data, we necessarily need to include those with low reliability, but they are often misdetections. Nevertheless, two approaches can be introduced as in the case of statistical estimation:

Minimization approach

We choose the value $\boldsymbol{\theta}$ that minimizes a specified cost function. This is regarded as the standard for computer vision applications.

Non-minimization approach

We compute the value $\boldsymbol{\theta}$ by solving a set of equations, which need not be defined by derivatives of some function. Hence, the solution does not necessarily minimize any cost function. As in the case of statistical estimation, this approach is more general and more flexible with a possibility of yielding better solutions than the minimization approach. However, this viewpoint is not well recognized in computer vision research.

2.6 KCR lower bound

For the minimization or the non-minimization approach, there exists a theoretical accuracy limit. This is formulated as follows. We assume that the true values $\bar{\boldsymbol{\xi}}_\alpha$ of the (transformed) observations $\boldsymbol{\xi}_\alpha$ satisfy the constraint $(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{\theta}) = 0$ for some $\boldsymbol{\theta}$. If we estimate it from the observations $\{\boldsymbol{\xi}_\alpha\}_{\alpha=1}^{N}$ by some means, the estimated value $\hat{\boldsymbol{\theta}}$ can be regarded as a

function $\hat{\boldsymbol{\theta}}(\{\boldsymbol{\xi}_\alpha\}_{\alpha=1}^N)$ of $\{\boldsymbol{\xi}_\alpha\}_{\alpha=1}^N$. This function is called an *estimator* of $\boldsymbol{\theta}$. Let $\Delta\boldsymbol{\theta}$ be its error, i.e., write $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$, and define the covariance matrix of $\hat{\boldsymbol{\theta}}$ by

$$V[\hat{\boldsymbol{\theta}}] = E[\Delta\boldsymbol{\theta}\Delta\boldsymbol{\theta}^\top], \tag{18}$$

where $E[\,\cdot\,]$ denotes expectation over data uncertainty. If we can assume that

- each $\boldsymbol{\xi}_\alpha$ is perturbed from its true value $\bar{\boldsymbol{\xi}}_\alpha$ by independent Gaussian noise of mean $\boldsymbol{0}$ and covariance matrix $V[\boldsymbol{\xi}_\alpha] = \sigma^2 V_0[\boldsymbol{\xi}_\alpha]$, and
- the function $\hat{\boldsymbol{\theta}}(\{\boldsymbol{\xi}_\alpha\}_{\alpha=1}^N)$ is an *unbiased estimator*, i.e., $E[\hat{\boldsymbol{\theta}}] = \boldsymbol{\theta}$ identically holds for whatever $\boldsymbol{\theta}$,

then the following inequality holds [5, 16, 17, 19].

$$V[\hat{\boldsymbol{\theta}}] \succ \frac{\sigma^2}{N}\Big(\frac{1}{N}\sum_{\alpha=1}^N \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}\Big)^-. \tag{19}$$

Here, $\boldsymbol{A} \succ \boldsymbol{B}$ means that $\boldsymbol{A}-\boldsymbol{B}$ is a positive semidefinite symmetric matrix, and $(\,\cdot\,)^-$ denotes the pseudo inverse. Chernov and Lesort [5] called the right side Eq. (19) the *Kanatani-Cramer-Rao* (*KCR*) *lower bound*. Equation (19) is for a single constraint in the form of Eq. (2) but can be extended to multiple constraints [24, 21] and to general nonlinear constraints [16].

3. Minimization Approach

First, we overview popular geometric estimation techniques for computer vision that are based on the minimization approach.

3.1 Least squares (LS)

Since the true values $\bar{\boldsymbol{\xi}}_\alpha$ of the observations $\boldsymbol{\xi}_\alpha$ satisfy $(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{\theta}) = 0$, we choose the value $\boldsymbol{\theta}$ that minimizes

$$J = \frac{1}{N}\sum_{\alpha=1}^N (\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2 \tag{20}$$

for noisy observations $\boldsymbol{\xi}_\alpha$ subject to the constraint $\|\boldsymbol{\theta}\| = 1$. This can also be viewed as minimizing $\sum_{\alpha=1}^N (\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2/\|\boldsymbol{\theta}\|^2$. Equation (20) can be rewritten in the form

$$\begin{aligned} J &= \frac{1}{N}\sum_{\alpha=1}^N (\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2 = \frac{1}{N}\sum_{\alpha=1}^N \boldsymbol{\theta}^\top \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top \boldsymbol{\theta} \\ &= (\boldsymbol{\theta}, \underbrace{\frac{1}{N}\sum_{\alpha=1}^N \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top}_{\equiv \boldsymbol{M}} \boldsymbol{\theta}) = (\boldsymbol{\theta}, \boldsymbol{M}\boldsymbol{\theta}), \end{aligned} \tag{21}$$

which is a quadratic form of $\boldsymbol{M}$. As is well known, the unit vector $\boldsymbol{\theta}$ that minimizes this quadratic form is given by the unit eigenvector of $\boldsymbol{M}$ for the smallest eigenvalue. Since the sum of squares is minimized, this method is is called *least squares* (*LS*). Equation (20) is often called the *algebraic distance*, so this method is also called *algebraic distance minimization*. Because the solution is directly obtained without any search, this method is widely used in many applications. However, it has been observed that the solution has large statistical bias. For ellipse fitting in *Example 2* (Sec. 2.2), for instance, the fitted ellipse is almost always smaller than the true shape. For this reason, this method is not suited for accurate estimation. However, this method is very convenient for rough estimation for guiding image processing, for the outlier-detection voting described in Sec. 2.2, and for initializing iterative optimization schemes.

3.2 Maximum likelihood (ML)

If the noise in each $\boldsymbol{x}_\alpha$ is an independent Gaussian variable of mean $\boldsymbol{0}$ and covariance matrix $V[\boldsymbol{x}_\alpha] = \sigma^2 V_0[\boldsymbol{x}_\alpha]$, the *Mahalanobis distance* of the observations $\{\boldsymbol{x}_\alpha\}$ from their true values $\{\bar{\boldsymbol{x}}_\alpha\}$ is

$$J = \frac{1}{N}\sum_{\alpha=1}^N (\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha, V_0[\boldsymbol{x}_\alpha]^{-1}(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha)), \tag{22}$$

and the likelihood of $\{\boldsymbol{x}_\alpha\}$ is written as $Ce^{-NJ/2\sigma^2}$, where C is a normalization constant that does not depend on $\bar{\boldsymbol{x}}_\alpha$ or $\boldsymbol{\theta}$. Thus, *maximum likelihood* (*ML*) that maximizes the likelihood is equivalent to minimizing Eq. (22) subject to the constraint

$$(\boldsymbol{\xi}(\bar{\boldsymbol{x}}_\alpha), \boldsymbol{\theta}) = 0. \tag{23}$$

As a special case, if the noise is homogeneous, i.e., independent of α, and isotropic, i.e., independent of the orientation, we can write $V_0[\boldsymbol{x}_\alpha] = \boldsymbol{I}$ (the identity), which reduces Eq. (22) to the *geometric distance*

$$J = \frac{1}{N}\sum_{\alpha=1}^N \|\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha\|^2. \tag{24}$$

Minimizing this subject to Eq. (23) is called *geometric distance minimization* by computer vision researchers and *total least squares* (*TLS*) by numerical analysis researchers[1]. If $\bar{\boldsymbol{x}}_\alpha$ represents the projection of the assumed 3-D structure onto the image plane and $\boldsymbol{x}_\alpha$ is its actually observed position, Eq. (24) is called the *reprojection error*. Minimizing it subject to Eq. (23) is often called *reprojection error minimization*.

Geometrically, ML can be interpreted to be fitting to N points $\boldsymbol{x}_\alpha$ in the data space the parameterized

[1] If the data $\boldsymbol{x}_\alpha$ are 2-D positions $\boldsymbol{x}_\alpha = (x_\alpha, y_\alpha)$, and if the y-coordinate alone undergoes noise, we only need to minimize $(1/N)\sum_{\alpha=1}^N (y_\alpha - \bar{y}_\alpha)^2$. In general, if only some components of the data $\boldsymbol{x}_\alpha$ contain noise, the problem is called *partial least squares* (*PLS*).

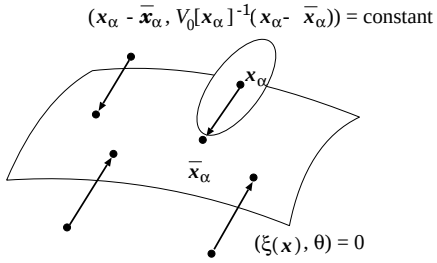


Figure 2: Fitting a hypersurface $(\boldsymbol{\xi}(\boldsymbol{x}), \boldsymbol{\theta}) = 0$ to points $\boldsymbol{x}_\alpha$ in the data space.

hypersurface $(\boldsymbol{\xi}(\boldsymbol{x}), \boldsymbol{\theta}) = 0$ by adjusting $\boldsymbol{\theta}$ (Fig. 2), where the discrepancy of the points from the surface is measured not by the Euclid distance but by the Mahalanobis distance of Eq. (22), which inversely weights the data by the covariances, thereby imposing heavier penalties on the points with higher certainty. In the field of computer vision, this approach is widely regarded as the ultimate method and often called the *Gold Standard* [12]. However, this is a highly nonlinear optimization problem and difficult to solve by a direct means. The difficulty stems from the fact that Eq. (23) is an implicit function of $\bar{\boldsymbol{x}}_\alpha$. If we could solve Eq. (23) for $\bar{\boldsymbol{x}}_\alpha$ to express it as an explicit function of $\boldsymbol{\theta}$, we could substitute it into Eq. (22) to obtain an unconstrained optimization problem for $\boldsymbol{\theta}$ alone, but this is generally not possible. In *Examples 1* (line fitting), *2* (ellipse fitting), and *3* (fundamental matrix computation) in Sec. 2.2, for instance, we cannot express (x, y) or (x, y, x', y') in terms of $\boldsymbol{\theta}$.

3.3 Bundle adjustment

A standard technique for minimizing Eq. (22) subject to Eq. (23) is to introduce a problem-dependent auxiliary variable to each $\boldsymbol{X}_\alpha$ and express $\bar{\boldsymbol{x}}_\alpha$ in terms of $\boldsymbol{X}_\alpha$ and $\boldsymbol{\theta}$ in the form

$$\bar{\boldsymbol{x}}_\alpha = \bar{\boldsymbol{x}}_\alpha(\boldsymbol{X}_\alpha, \boldsymbol{\theta}). \tag{25}$$

Then, we substitute this into Eq. (22) and minimize

$$J(\{\boldsymbol{X}_\alpha\}_{\alpha=1}^N, \boldsymbol{\theta}) = \frac{1}{N}\sum_{\alpha=1}^{N}(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha(\boldsymbol{X}_\alpha, \boldsymbol{\theta}), V_0[\boldsymbol{x}_\alpha]^{-1}(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha(\boldsymbol{X}_\alpha, \boldsymbol{\theta}))) \tag{26}$$

over the joint parameter space of $\{\boldsymbol{X}_\alpha\}_{\alpha=1}^N$ and $\boldsymbol{\theta}$.

A typical example of this approach is 3-D reconstruction from multiple images (Fig. 3), for which $\boldsymbol{x}_\alpha$ has the form of $\boldsymbol{x}_\alpha = (x_\alpha, y_\alpha, x'_\alpha, y'_\alpha, ..., x''_\alpha, y''_\alpha)$, concatenating the projections (x_α, y_α), (x'_α, y'_α), ..., (x''_α, y''_α) of the αth point in the scene onto the images. The unknown parameter $\boldsymbol{\theta}$ specifies the state of all the cameras, consisting of the extrinsic parameters (the positions and orientations) and the intrinsic parameters (the focal lengths, the principal points, the aspect ratios, and the skew angles). If we introduce the

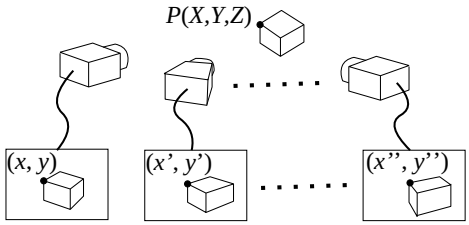


Figure 3: 3-D reconstruction by bundle adjustment

3-D position $\boldsymbol{X}_\alpha = (X_\alpha, Y_\alpha, Z_\alpha)$ of each point in the scene as the auxiliary variable, the true value $\bar{\boldsymbol{x}}_\alpha$ of $\boldsymbol{x}_\alpha$ can be explicitly expressed in the form $\bar{\boldsymbol{x}}_\alpha(\boldsymbol{X}_\alpha, \boldsymbol{\theta})$, which describes the image positions of the 3-D point $\boldsymbol{X}_\alpha$ that should be observed if the cameras have the parameter $\boldsymbol{\theta}$. Then, we minimize the *reprojection error*, i.e., the discrepancy of the observed projections $\boldsymbol{x}_\alpha$ from the predicted projections $\bar{\boldsymbol{x}}_\alpha(\boldsymbol{X}_\alpha, \boldsymbol{\theta})$. The minimum is searched over the entire parameter space of $\{\boldsymbol{X}_\alpha\}_{\alpha=1}^N$ and $\boldsymbol{\theta}$. This process is called *bundle adjustment* [30, 32, 44], a term originated from photogrammetry, meaning we "adjust" the "bundle" of lines of sight so that they pass through the observed points in images. The package program is available on the Web [32]. The dimension of the parameter space is $3N$ + "the dimension of $\boldsymbol{\theta}$", which becomes very large when many points are observed.

This bundle adjustment approach is not limited to 3-D reconstruction from multiple images. In *Examples 1* (line fitting) and *2* (ellipse fitting) in Sec. 2.2, for example, if we introduce the arc length s_α of the true position $(\bar{x}_\alpha, \bar{y}_\alpha)$ along the line or the ellipse from a fixed reference point as the auxiliary variable, we can express (x_α, y_α) in terms of s_α and $\boldsymbol{\theta}$. Then, we minimize the resulting Mahalanobis distance J over the entire parameter space of s_1, ..., s_N and $\boldsymbol{\theta}$. Instead of the arc length s_α, we can alternatively use the argument ϕ_α measured from the x-axis [42]. A similar approach can be done for fundamental matrix computation [4].

The standard numerical technique for the search of the parameter space is the *Levenberg-Marquardt* (*LM*) *method* [38], which is a hybrid of the Gauss-Newton iterations and the gradient descent. However, depending on the initial value of the iterations, the search may fall into a local minimum. Various global optimization techniques have also been studied [11]. A typical method is *branch and bound*, which introduces a function that gives a lower bound of J over a given region and divides the parameter space into small cells; those cells which have lower bounds that are above the tested values are removed, and other cells are recursively subdivided [11, 14]. However, the evaluation of the lower bound involves a complicated technique, and searching the entire space requires a significant amount of computational time.

3.4 Nuisance parameters

As we can see from Eq. (26), the number of unknowns increases as we observe more data $\boldsymbol{x}_\alpha$; there exist as many auxiliary variables as the number of data. As pointed out in Sec. 2.4, the asymptotic analysis of $N \to \infty$ for the number N of observations is a major concern in the domain of standard statistical estimation, but the analysis becomes anomalous if the number of unknowns also increases as $N \to \infty$. For this reason, the auxiliary variables $\boldsymbol{X}_\alpha$ are called *nuisance parameters*, while $\boldsymbol{\theta}$ is called the *structural parameter* or the *parameter of interest*. The fact that the standard asymptotic properties of ML for $N \to \infty$ do not hold in the presence of nuisance parameters was pointed out by Neyman and Scott [34] in 1940s, and this anomaly is known as the *Neyman-Scott problem*. Although the asymptotic analysis of $N \to \infty$ does not make much sense for computer vision applications as pointed out in Sec. 2.5, this is a serious problem for standard statistical estimation based on repeated sampling. One mathematical technique for avoiding this anomaly is to regard the auxiliary variables $\boldsymbol{X}_\alpha$ not as unknowns but as "samples" from an unknown probability density $g(\boldsymbol{X})$ that does not depend on N; we estimate the function $g(\boldsymbol{X})$ and the parameter $\boldsymbol{\theta}$ simultaneously. This approach is called the *semiparametric model*. [2, 3]. Okatani and Deguchi [35] applied this approach to 3-D reconstruction from multiple images.

3.5 Gaussian approximation in the $\boldsymbol{\xi}$-space

The search in a high-dimensional parameter space of the bundle adjustment approach can be avoided if we introduce Gaussian approximation to the noise distribution in the $\boldsymbol{\xi}$-space. As pointed out in Sec. 2.3, if the noise in the observation $\boldsymbol{x}_\alpha$ is Gaussian, the noise in its nonlinear mapping $\boldsymbol{\xi}_\alpha = \boldsymbol{\xi}(\boldsymbol{x}_\alpha)$ is not strictly Gaussian, although it is expected to have a Gaussian-like distribution if the noise is small. If it is approximated to be Gaussian, the optimization computation becomes much simpler. Suppose $\boldsymbol{\xi}_\alpha$ has noise of mean $\boldsymbol{0}$ with the covariance matrix $V[\boldsymbol{\xi}_\alpha] = \sigma^2 V_0[\boldsymbol{\xi}_\alpha]$ evaluated by Eq. (13). Then, the ML computation reduces to minimizing the Mahalanobis distance

$$J = \frac{1}{N}\sum_{\alpha=1}^{N}(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha]^{-1}(\boldsymbol{\xi}_\alpha - \bar{\boldsymbol{\xi}}_\alpha)) \tag{27}$$

in the $\boldsymbol{\xi}$-space subject to the *linear* constraint

$$(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{\theta}) = 0. \tag{28}$$

Geometrically, this is interpreted to be fitting to N points $\boldsymbol{\xi}_\alpha$ in the $\boldsymbol{\xi}$-space the parameterized "hyperplane" $(\boldsymbol{\xi}, \boldsymbol{\theta}) = 0$ by adjusting $\boldsymbol{\theta}$, where the discrepancy of the points from the plane is measured by the Mahalanobis distance of Eq. (27) inversely weighted by the covariances of the data in the $\boldsymbol{\xi}$-space (Fig. 4). Since Eq. (28) is now "linear" in $\bar{\boldsymbol{\xi}}_\alpha$, this constraint can be eliminated using Lagrange multipliers, reducing the problem to unconstrained minimization of

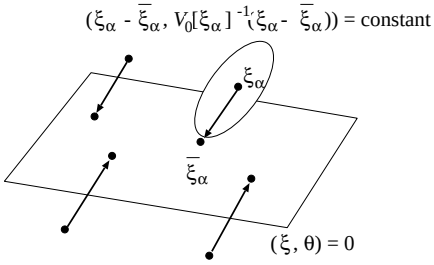


Figure 4: Fitting a hyperplane $(\boldsymbol{\xi}, \boldsymbol{\theta}) = 0$ to points $\boldsymbol{\xi}_\alpha$ in the $\boldsymbol{\xi}$-space.

$$J = \frac{1}{N}\sum_{\alpha=1}^{N}\frac{(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}. \tag{29}$$

This means, in the statistical terms of Sec. 3.4, we have effectively eliminated the nuisance parameters $\bar{\boldsymbol{\xi}}_\alpha$. Today, Eq. (29) is called the *Sampson error* [12] after the ellipse fitting scheme introduced by P. D. Sampson [41].

3.6 Sampson error minimization

Various numerical techniques have been proposed for minimizing the Sampson error in Eq. (29). The best known is the *FNS* (*Fundamental Numerical Scheme*) of Chojnacki et al. [7], which goes as follows:

1. Let $W_\alpha = 1$, $\alpha = 1, ..., N$, and $\boldsymbol{\theta}_0 = \boldsymbol{0}$.
2. Comput the matrices

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top,$$
$$\boldsymbol{L} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha^2 (\boldsymbol{\theta}_0, \boldsymbol{\xi}_\alpha)^2 V_0[\boldsymbol{\xi}_\alpha]. \tag{30}$$

3. Solve the eigenvalue problem

$$(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{\theta} = \lambda\boldsymbol{\theta}, \tag{31}$$

   and compute the unit eigenvector $\boldsymbol{\theta}$ for the smallest[2] eigenvalue λ.
4. If $\boldsymbol{\theta} \approx \boldsymbol{\theta}_0$ up to sign, return $\boldsymbol{\theta}$ and stop. Else, let

$$W_\alpha \leftarrow \frac{1}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}, \quad \boldsymbol{\theta}_0 \leftarrow \boldsymbol{\theta}, \tag{32}$$

   and go back to Step 2.

[2]We can alternatively compute the unit eigenvector $\boldsymbol{\theta}$ for the smallest eigenvalue λ in absolute value, but it has been experimentally confirmed that convergence is faster for computing the smallest eigenvalue [26].

The background of this method is as follows. At the time of convergence, the matrices $\boldsymbol{M}$ and $\boldsymbol{L}$ have the form

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N}\frac{\boldsymbol{\xi}_\alpha\boldsymbol{\xi}_\alpha^\top}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})},$$
$$\boldsymbol{L} = \frac{1}{N}\sum_{\alpha=1}^{N}\frac{(\boldsymbol{\theta}, \boldsymbol{\xi}_\alpha)^2 V_0[\boldsymbol{\xi}_\alpha]}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})^2}. \quad (33)$$

It is easily seen that the derivative of the Sampson error J in Eq. (29) is written in terms of these matrices in the form

$$\nabla_{\boldsymbol{\theta}} J = 2(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{\theta}. \quad (34)$$

We can prove that the FNS iterations has a unique fixed point. To be specific, when the iterations have converged, the eigenvalue λ in Eq. (31) is shown to be 0. Hence, the value $\boldsymbol{\theta}$ returned by the above procedure is the solution of $\nabla_{\boldsymbol{\theta}} J = \boldsymbol{0}$.

Other methods exist for minimizing Eq. (29) including the *HEIV* (*Heteroscedastic Errors-in-Variables*) of Leedan and Meer [31] and Matei and Meer [33], and the *projective Gauss-Newton iterations* of Kanatani and Sugaya [26], all computing the same solution. Note that the "initial solution" obtained in the beginning by letting $W_\alpha = 1$ coincides with the LS solution described in Sec. 3.1. The above procedure is for the case of a single constraint in the form of Eq. (2) but can be straightforwardly extended to the case of multiple constraints [21].

3.7 Minimization with internal constraints

In some applications, the parameter $\boldsymbol{\theta}$ may have *internal constraints* in the form of $\phi_1(\boldsymbol{\theta}) = 0$, ..., $\phi_r(\boldsymbol{\theta}) = 0$, where $\phi_1(\boldsymbol{\theta})$, ..., $\phi_r(\boldsymbol{\theta})$ are *homogeneous polynomials*.

Example 4 Suppose the matrix $\boldsymbol{F} = \begin{pmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6 \\ \theta_7 & \theta_8 & \theta_9 \end{pmatrix}$ is constrained to have determinant 0. The internal constraint $\det \boldsymbol{F} = 0$ is written as $\phi(\boldsymbol{F}) = 0$, where

$$\phi(\boldsymbol{\theta}) = \theta_1\theta_5\theta_6 + \theta_2\theta_6\theta_7 + \theta_3\theta_8\theta_4 - \theta_3\theta_5\theta_7 - \theta_2\theta_4\theta_9 - \theta_1\theta_8\theta_6. \quad (35)$$

Example 5 Suppose the matrix $\boldsymbol{R} = \begin{pmatrix} \theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6 \\ \theta_7 & \theta_8 & \theta_9 \end{pmatrix}$ is constrained to be a rotation. The internal constraint $\boldsymbol{R}^\top\boldsymbol{R} = \boldsymbol{I}$ is written as $\phi_1(\boldsymbol{R}) = 0$, ..., $\phi_6(\boldsymbol{R}) = 0$, where

$$\begin{aligned}
\phi_1(\boldsymbol{\theta}) &= \theta_1\theta_4 + \theta_2\theta_5 + \theta_3\theta_6, \\
\phi_2(\boldsymbol{\theta}) &= \theta_4\theta_7 + \theta_5\theta_8 + \theta_6\theta_9, \\
\phi_3(\boldsymbol{\theta}) &= \theta_7\theta_1 + \theta_8\theta_2 + \theta_9\theta_3, \\
\phi_4(\boldsymbol{\theta}) &= \theta_1^2 + \theta_2^2 + \theta_3^2 - \theta_4^2 - \theta_5^2 - \theta_6^2, \\
\phi_5(\boldsymbol{\theta}) &= \theta_4^2 + \theta_5^2 + \theta_6^2 - \theta_7^2 - \theta_8^2 - \theta_9^2, \\
\phi_6(\boldsymbol{\theta}) &= \theta_1^2 + \theta_2^2 + \theta_3^2 - \theta_0. \quad (36)
\end{aligned}$$

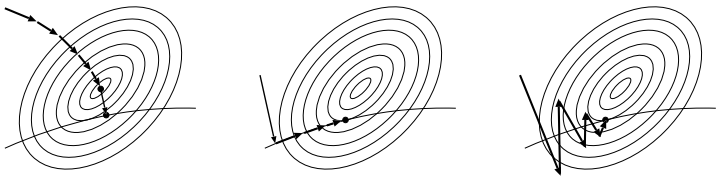
Here, θ_0 is a dummy parameter. In the end, we $\boldsymbol{\theta}$ is re-scaled so that $\theta_0 = 1$ in the end.

The *EFNS* (*Extended FNS*) of Kanatani and Sugaya [25, 28] for minimizing the Sampson error in the presence of internal constraints go as follows:

1. Provide an initial guess $\boldsymbol{\theta}$.
2. Compute the matrices
$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha\boldsymbol{\xi}_\alpha\boldsymbol{\xi}_\alpha^\top,$$
$$\boldsymbol{L} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha^2(\boldsymbol{\theta}_0, \boldsymbol{\xi}_\alpha)^2 V_0[\boldsymbol{\xi}_\alpha]. \quad (37)$$
3. Compute $\nabla_{\boldsymbol{\theta}}\phi_1(\boldsymbol{\theta})$, ..., $\nabla_{\boldsymbol{\theta}}\phi_r(\boldsymbol{\theta})$. Let $\{\boldsymbol{v}_1, ..., \boldsymbol{v}_r\}$ be the orthonormal system obtained from them by Schmidt orthogonalization, and compute the projection matrix
$$\boldsymbol{P}_{\mathcal{M}} = \boldsymbol{I} - \sum_{k=1}^{r}\boldsymbol{v}_k\boldsymbol{v}_k^\top. \quad (38)$$
4. Let $\boldsymbol{v}_0$, ..., $\boldsymbol{v}_r$ be the unit eigenvectors of
$$\boldsymbol{X} = \boldsymbol{P}_{\mathcal{M}}(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{P}_{\mathcal{M}} \quad (39)$$
for the smallest $r+1$ eigenvalues.
5. Project the current value $\boldsymbol{\theta}$ onto the linear space spanned by $\{\boldsymbol{\theta}_0, ..., \boldsymbol{\theta}_r\}$:
$$\hat{\boldsymbol{\theta}} = \sum_{k=0}^{r}(\boldsymbol{\theta}, \boldsymbol{\theta}_k)\boldsymbol{\theta}_k. \quad (40)$$
6. Compute
$$\boldsymbol{\theta}' = \mathcal{N}[\boldsymbol{P}_{\mathcal{M}}\hat{\boldsymbol{\theta}}], \quad (41)$$
where $\mathcal{N}[\,\cdot\,]$ denotes normalization to unit norm ($\mathcal{N}[\boldsymbol{a}] \equiv \boldsymbol{a}/\|\boldsymbol{a}\|$).
7. If $\boldsymbol{\theta} \approx \boldsymbol{\theta}_0$ up to sign, return $\boldsymbol{\theta}$ and stop. Else, let
$$W_\alpha \leftarrow \frac{1}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}, \quad \boldsymbol{\theta}_0 \leftarrow \mathcal{N}[\boldsymbol{\theta} + \boldsymbol{\theta}'], \quad (42)$$
and go back to Step 2.

As in the case of FNS, the fixed point of the EFNS iterations is proved to be unique [25, 28]. Hence, when the iterations have converged, the returned value necessarily satisfies $\phi_1(\boldsymbol{\theta}) = 0$, ..., $\phi_r(\boldsymbol{\theta}) = 0$ and stationalizes J.

The EFNS of Kanatani and Sugaya [25, 28] was devised to compute the fundamental matrix $\boldsymbol{F}$, which is constrained to have rank 2, for 3-D reconstruction from two views. The procedure is straightforwardly extended to multiple constraints as in the case of FNS. In the past, minimization in the presence of internal constraints has been done in many different ways. They are roughly classified into three categories:



(a) (b) (c)

Figure 5: (a) A posteriori correction. (b) Internal access. (c) External access.

A posteriori correction: Minimize J without considering the internal constraints and modify the solution $\boldsymbol{u}$ so that they are satisifed (Fig. 5(a)).

Internal access: Parameterize $\boldsymbol{\theta}$ so that internal constraints are identically satisfied and minimize J in the resulting ("internal") parameter space (Fig. 5(b)).

External access: Do iterations in the unconstrained ("external") parameter space in such a way that the solution automatically satisfies the internal constraints at the time of convergence (Fig. 5(c)).

EFNS is an external access approach.

3.8 Computation of the exact ML solution

Since the Sampson error of Eq. (29) is obtained by approximating the non-Gaussian noise distribution in the $\boldsymbol{\xi}$-space by a Gaussian distribution, the solution does not necessarily coincide with the ML solution that minimizes the Mahalanobis distance in Eq. (22). However, once we have obtained the solution $\boldsymbol{\theta}$ that minimizes Eq. (29), we can iteratively modify Eq. (29) by using that $\boldsymbol{\theta}$ so that Eq. (29) coincides with Eq. (22) in the end. This means that we obtain the exact ML solution. The procedure goes as follows [29]:

1. Let $J_0^* = \infty$ (a sufficiently large number), $\hat{\boldsymbol{x}}_\alpha = \boldsymbol{x}_\alpha$, and $\tilde{\boldsymbol{x}}_\alpha = \boldsymbol{0}$, $\alpha = 1, ..., N$.
2. Evaluate the normalized covariance matrices $V_0[\hat{\boldsymbol{\xi}}_\alpha]$ by replacing $\boldsymbol{x}_\alpha$ by $\hat{\boldsymbol{x}}_\alpha$ in their definition.
3. Compute the following $\boldsymbol{\xi}_\alpha^*$:
$$\boldsymbol{\xi}_\alpha^* = \boldsymbol{\xi}_\alpha + \left.\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_\alpha} \tilde{\boldsymbol{x}}_\alpha. \tag{43}$$
4. Compute the value $\boldsymbol{\theta}$ that minimizes the *modified Sampson error*
$$J^* = \frac{1}{N}\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha^*, \boldsymbol{\theta})^2}{(\boldsymbol{\theta}, V_0[\hat{\boldsymbol{\xi}}_\alpha]\boldsymbol{\theta})}. \tag{44}$$
5. Update $\tilde{\boldsymbol{x}}_\alpha$ and $\hat{\boldsymbol{x}}_\alpha$ as follows:
$$\tilde{\boldsymbol{x}}_\alpha \leftarrow \frac{(\boldsymbol{\xi}_\alpha^*, \boldsymbol{\theta}) V_0[\boldsymbol{x}_\alpha]}{(\boldsymbol{\theta}, V_0[\hat{\boldsymbol{\xi}}_\alpha]\boldsymbol{\theta})} \left.\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}_\alpha}^{\top} \boldsymbol{\theta},$$
$$\hat{\boldsymbol{x}}_\alpha \leftarrow \boldsymbol{x}_\alpha - \tilde{\boldsymbol{x}}_\alpha. \tag{45}$$
6. Evaluate J^* by
$$J^* = \frac{1}{N}\sum_{\alpha}^{N} (\tilde{\boldsymbol{x}}_\alpha, V_0[\boldsymbol{x}_\alpha]\tilde{\boldsymbol{x}}_\alpha). \tag{46}$$
If $J^* \approx J_0$, return $\boldsymbol{\theta}$ and stop. Else, let $J_0 \leftarrow J^*$ and go back to Step 2.

Since the modified Sampson error in Eq. (44) has the same form as the Sampson error in Eq. (29), we can minimize it by FNS (or HEIV or projective Gauss-Newton iterations). According to numerical experiments, this iterative modification converges after four or five rounds, yet in many practical problems the first four or five effective figures remain unchanged [21, 27, 28]. In this sense, we can practically identify the Sampson error minimization with the ML computation.

3.9 Hyperaccurate correction

It has been widely recognized that the Sampson error minimization solution, which can be practically identified with the ML solution, has very high accuracy. However, it can be shown by detailed error analysis that the solution has statistical bias of $O(\sigma^2)$ and that the magnitude of the bias can be theoretically evaluated [19]. This implies that the accuracy can be further improved by subtracting the theoretically expected bias. This process is called *hyperaccurate correction* and goes as follows [18, 19]:

1. Estimate the square noise level σ^2 from the computed solution $\boldsymbol{\theta}$ and the corresponding matrix $\boldsymbol{M}$ in Eq. (33) by
$$\hat{\sigma}^2 = \frac{(\boldsymbol{\theta}, \boldsymbol{M}\boldsymbol{\theta})}{1 - (n-1)/N}, \tag{47}$$
where n is the dimension of the vector $\boldsymbol{\theta}$.
2. Compute the correction term[3]
$$\Delta_c\boldsymbol{\theta} = -\frac{\sigma^2}{N}\boldsymbol{M}_{n-1}^{-}\sum_{\alpha=1}^{N} W_\alpha(\boldsymbol{e}_\alpha, \boldsymbol{\theta})\boldsymbol{\xi}_\alpha + \frac{\hat{\sigma}^2}{N^2}\boldsymbol{M}_{n-1}^{-}\sum_{\alpha=1}^{N} W_\alpha^2(\boldsymbol{\xi}_\alpha, \boldsymbol{M}_{n-1}^{-}V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})\boldsymbol{\xi}_\alpha, \tag{48}$$

[3]The first term of Eq. (48) is omitted in [18, 19].

where $\boldsymbol{e}_\alpha$ is a vector that depends on individual problems, and $\boldsymbol{M}_{n-1}^-$ is the pseudoinverse of $\boldsymbol{M}$ with truncated rank $n-1$ computed after the smallest eigenvalue is replaced by 0 in its spectral decomposition.

3. Correct the solution $\boldsymbol{\theta}$ in the form

$$\boldsymbol{\theta} \leftarrow \mathcal{N}[\boldsymbol{\theta} - \Delta_c\boldsymbol{\theta}], \tag{49}$$

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

The vector $\boldsymbol{e}_\alpha$ is $\boldsymbol{0}$ for many problems including line fitting (*Example 1* in Sec. 2.2) and fundamental matrix computation (*Example 3* in Sec. 2.2). It is generally $\boldsymbol{0}$ if multiple images are involved. A typical problem of nonzero $\boldsymbol{e}_\alpha$ is ellipse fitting (*Example 2* in Sec. 2.2), for which $\boldsymbol{e}_\alpha = (1,0,1,0,0,0)^\top$. However, the effect is negligibly small, and the solution is practically the same if $\boldsymbol{e}_\alpha$ is replaced by $\boldsymbol{0}$.

The above bias correction concerns geometric estimation based on the geometric model of Eq. (2). In statistics, on the other hand, it is known that ML entails statistical bias in the presence of nuisance parameters, and various studies exist for analyzing and removing bias in the ML solution. Okatani and Deguchi [36, 37] applied them to vision problems by introducing auxiliary variables in the form of Eq. (25). They analyzed the relationship between the bias and the hypersurface defined by the constraint [36] and introduced the method of projected scores [37].

Since ML, of which reprojection error minimization is a special case, has been commonly regarded as the ultimate method (*Gold Standard* [12]) for computer vision, the fact that the accuracy of the ML solution can be further improved by the above hyperaccurate correction is surprising to many vision researchers. For hyperaccurate correction, however, one first needs to obtain the ML solution by an iterative method such as FNS and also estimate the noise level σ. Then, a question arises. Is it not possible to directly compute the corrected solution from the beginning, say, by modifying the FNS iterations? We now show that this is possible if we adopt the non-minimization approach of geometric estimation.

4. Non-minimization Approach

4.1 Iterative reweight

The oldest method that is not based on minimization is the following *iterative reweight*:

1. Let $W_\alpha = 1$, $\alpha = 1, ..., N$, and $\boldsymbol{\theta}_0 = \boldsymbol{0}$.
2. Comput the following matrix $\boldsymbol{M}$:

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top. \tag{50}$$

3. Solve the eigenvalue problem

$$\boldsymbol{M}\boldsymbol{\theta} = \lambda\boldsymbol{\theta}, \tag{51}$$

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

4. If $\boldsymbol{\theta} \approx \boldsymbol{\theta}_0$ up to sign, return $\boldsymbol{\theta}$ and stop. Else, let

$$W_\alpha \leftarrow \frac{1}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}, \qquad \boldsymbol{\theta}_0 \leftarrow \boldsymbol{\theta}, \tag{52}$$

and go back to Step 2.

The motivation of this method is the *weighted least squares* that minimizes

$$\begin{aligned}\frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2 &= \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\theta}^\top \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top \boldsymbol{\theta} \\ &= (\boldsymbol{\theta}, \underbrace{\frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top}_{\equiv \boldsymbol{M}} \boldsymbol{\theta}) = (\boldsymbol{\theta}, \boldsymbol{M}\boldsymbol{\theta}).\end{aligned} \tag{53}$$

This is minimized by the unit eigenvector $\boldsymbol{\theta}$ of the matrix $\boldsymbol{M}$ for the smallest eigenvalue. As is well known in statistics, the optimal choice of the weight W_α is the inverse of the variance of that term. Since $(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{\theta}) = 0$, we have $(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta}) = (\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{\theta}) + \cdots$, and hence the leading term of the variance is

$$\begin{aligned}E[(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2] &= E[\boldsymbol{\theta}^\top \Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top \boldsymbol{\theta}] \\ &= (\boldsymbol{\theta}, E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top]\boldsymbol{\theta}) = \sigma^2(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta}).\end{aligned} \tag{54}$$

Hence, we should choose

$$W_\alpha = \frac{1}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}, \tag{55}$$

but $\boldsymbol{\theta}$ is unknown. So, we do iterations, determining the weight W_α from the value of $\boldsymbol{\theta}$ in the preceding step. The "initial solution" computed with $W_\alpha = 1$ coincides with the LS solution, minimizing Eq. (20) in Sec. 3.1.

If Eq. (55) is substituted, Eq. (53) coincides with the Sampson error in Eq. (29). With the iterative update in Eq. (52), it appears that Eq. (29) is minimized. However, we are computing at each step the value of $\boldsymbol{\theta}$ that minimizes the numerator part for the fixed value of the denominator terms determined in the preceding step. Hence, at the time of the convergence, the resulting solution $\boldsymbol{\theta}$ is such that

$$\frac{1}{N}\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})} \leq \frac{1}{N}\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta}')^2}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})} \tag{56}$$

for any $\boldsymbol{\theta}'$, but this does not mean

$$\frac{1}{N}\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta})^2}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})} \leq \frac{1}{N}\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \boldsymbol{\theta}')^2}{(\boldsymbol{\theta}', V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta}')}. \tag{57}$$

The fact that iterative reweight does not minimize a particular cost function has not been well recognized by vision researchers.

The perturbation analysis in [19] shows that the covariance matrix $V[\boldsymbol{\theta}]$ of the resulting solution $\boldsymbol{\theta}$ agrees with the KCR lower bound (Sec. 2.6) up to $O(\sigma^4)$. Hence, it is practically impossible to reduce the variance any further. However, it has been widely known that the iterative reweight solution has large bias [16]. Thus, the following strategies were introduced to improve iterative reweight:

- Remove the bias of the solution.
- Exactly minimize the Sampson error in Eq. (29).

The former is Kanatani's renormalization [15, 16], and the latter is the FNS of Chojnacki et al. [7] and the HEIV of Leedan and Meer [31] and Matei and Meer [33].

4.2 Renormalization

Kanatani's renormalization [15, 16] goes as follows[4]:

1. Let $W_\alpha = 1$, $\alpha = 1, ..., N$, and $\boldsymbol{\theta}_0 = \boldsymbol{0}$.
2. Compute the following matrices $\boldsymbol{M}$ and $\boldsymbol{N}$:
$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top, \qquad \boldsymbol{N} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha V_0[\boldsymbol{\xi}_\alpha]. \tag{58}$$
3. Solve the generalized eigenvalue problem
$$\boldsymbol{M}\boldsymbol{\theta} = \lambda \boldsymbol{N}\boldsymbol{\theta}, \tag{59}$$
and compute the unit eigenvector $\boldsymbol{\theta}$ for the smallest eigenvalue λ in absolute value.
4. If $\boldsymbol{\theta} \approx \boldsymbol{\theta}_0$ up to sign, return $\boldsymbol{\theta}$ and stop. Else, let
$$W_\alpha \leftarrow \frac{1}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}, \qquad \boldsymbol{\theta}_0 \leftarrow \boldsymbol{\theta}, \tag{60}$$
and go back to Step 2.

The motivation of renormalization is as follows. Let $\bar{\boldsymbol{M}}$ be the true value of the matrix $\boldsymbol{M}$ in Eq. (58). Since $(\bar{\boldsymbol{\xi}}_\alpha, \boldsymbol{\theta}) = 0$, we have $\bar{\boldsymbol{M}}\boldsymbol{\theta} = \boldsymbol{0}$. Hence, $\boldsymbol{\theta}$ is the eigenvector of $\bar{\boldsymbol{M}}$ for eigenvalue 0, but $\bar{\boldsymbol{M}}$ is unknown. So, we estimate it. Since $E[\Delta\boldsymbol{\xi}_\alpha] = 0$ to a first approximation, the expectation of $\boldsymbol{M}$ is

$$\begin{aligned} E[\boldsymbol{M}] &= E[\frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha (\bar{\boldsymbol{\xi}}_\alpha + \Delta\boldsymbol{\xi}_\alpha)(\bar{\boldsymbol{\xi}}_\alpha + \Delta\boldsymbol{\xi}_\alpha)^\top] \\ &= \bar{\boldsymbol{M}} + \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top] \\ &= \bar{\boldsymbol{M}} + \frac{\sigma^2}{N}\sum_{\alpha=1}^{N} W_\alpha V_0[\boldsymbol{\xi}_\alpha] = \bar{\boldsymbol{M}} + \sigma^2 \boldsymbol{N}. \end{aligned} \tag{61}$$

Thus, $\bar{\boldsymbol{M}} = E[\boldsymbol{M}] - \sigma^2\boldsymbol{N} \approx \boldsymbol{M} - \sigma^2\boldsymbol{N}$, so instead of $\bar{\boldsymbol{M}}\boldsymbol{\theta} = \boldsymbol{0}$ we solve $(\boldsymbol{M} - \sigma^2\boldsymbol{N})\boldsymbol{\theta} = \boldsymbol{0}$, or $\boldsymbol{M}\boldsymbol{\theta} = \sigma^2\boldsymbol{N}\boldsymbol{\theta}$. Assuming that σ^2 is small, we regard it as the smallest eigenvalue λ in absolute value. As in the case of iterative reweight, we iteratively update the weight W_α so that it approaches Eq. (55). The above procedure is for the case of single constraint in the form of Eq. (2) but can be straightforwardly extended to the case of multiple constraints [16, 22].

Kanatani's renormalization [15, 16] attracted much attention because it exhibited higher accuracy than any other then known methods. However, questions were repeatedly raised as to what it minimizes, perhaps out of the deep-rooted preconception that optimal estimation should minimize something. Chojnacki et al. [6] argued that renormalization can be "rationalized" if viewed as approximately minimizing the Sampson error. However, the renormalization process is not minimizing any particular cost function.

Note that the initial solution with $W_\alpha = 1$ solves $\Big(\sum_{\alpha=1}^{N} \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top\Big)\boldsymbol{\theta} = \lambda\Big(\sum_{\alpha=1}^{N} V_0[\boldsymbol{\xi}_\alpha]\Big)\boldsymbol{\theta}$, which is nothing but the method of Taubin [43], known to be very accurate algebraic method without requiring iterations. Thus, *renormalization is an iterative improvement of the Taubin solution*. According to many experiments, renormalization is shown to be more accurate than the Taubin method with nearly comparable accuracy with the FNS and the HEIV. The accuracy of renormalization is analytically evaluated in [19], showing that the covariance matrix $V[\boldsymbol{\theta}]$ of the solution $\boldsymbol{\theta}$ agrees with the KCR lower bound up to $O(\sigma^4)$ just as iterative reweight, but the bias is much smaller. That is the reason for the high accuracy of renormalization. According to experiences, the estimation error has roughly the following order:

LS > iterative reweight > Taubin > renormalization > ML.

4.3 Analysis of covariance and bias

Since the covariance matrix $V[\boldsymbol{\theta}]$ of the renormalization solution $\boldsymbol{\theta}$ agrees with the KCR lower bound up to $O(\sigma^4)$, the covariance of the solution cannot be substantially improved any further. Very small it may

[4] This is slightly different from the description in [15], in which the generalized eigenvalue problem is reduced to the standard eigenvalue problem, but the resulting solution is the same [16].

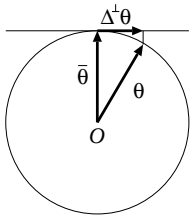


Figure 6: The true value $\bar{\boldsymbol{\theta}}$, the computed value $\boldsymbol{\theta}$, and its orthogonal component $\Delta^{\perp}\boldsymbol{\theta}$ of $\bar{\boldsymbol{\theta}}$.

be, however, the bias is not 0. Note that the renormalization procedure reduces to iterative reweight if the matrix $\boldsymbol{N}$ in Eq. (59) is replaced by the identity $\boldsymbol{I}$. This means that the reduction of the bias is attributed to the matrix $\boldsymbol{N}$ in Eq. (59). This observation implies the possibility of further reducing the bias by *optimizing* the matrix $\boldsymbol{N}$ in the form

$$\boldsymbol{N} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha V_0[\boldsymbol{\xi}_\alpha] + \cdots, \tag{62}$$

so that *the bias is zero up to high order error terms*. Using the perturbation analysis in [19], Al-Sharadqah and Chernov [1] actually did this for ellipse fitting, and Kanatani et al. [20] extended it to general geometric estimation. Their analysis goes as follows.

We write the observation $\boldsymbol{x}_\alpha$ as the sum $\boldsymbol{x}_\alpha = \bar{\boldsymbol{x}}_\alpha + \Delta\boldsymbol{x}_\alpha$ of the true value $\bar{\boldsymbol{x}}_\alpha$ and the noise term $\Delta\boldsymbol{x}_\alpha$. Substituting this into $\boldsymbol{\xi}_\alpha = \boldsymbol{\xi}(\boldsymbol{x}_\alpha)$ and expand it in the form

$$\bar{\boldsymbol{\xi}}_\alpha + \Delta_1\boldsymbol{\xi}_\alpha + \Delta_2\boldsymbol{\xi}_\alpha + \cdots, \tag{63}$$

where and hereafter the bar denotes the noiseless value and Δ_k denotes terms of $O(\sigma^k)$. We similarly expand $\boldsymbol{M}$, $\boldsymbol{\theta}$, $\boldsymbol{\lambda}$, and $\boldsymbol{N}$ and express Eq. (59) in the form

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

Equating the terms of the same order in σ, we obtain

$$\Delta_1\boldsymbol{\theta} = -\bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}\bar{\boldsymbol{\theta}}, \tag{64}$$

$$\Delta_2^{\perp}\boldsymbol{\theta} = \bar{\boldsymbol{M}}^{-}\Big(\frac{(\bar{\boldsymbol{\theta}}, \boldsymbol{T}\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}})}\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}} - \boldsymbol{T}\bar{\boldsymbol{\theta}}\Big), \tag{65}$$

where $\bar{\boldsymbol{M}}^{-}$ is the pseudoinverse of $\bar{\boldsymbol{M}}$; since $\bar{\boldsymbol{M}}$ has the eigenvector $\bar{\boldsymbol{\theta}}$ of eigenvalue 0, it has rank $n-1$ (n is the dimension of $\boldsymbol{\theta}$). The symbol $\Delta_2^{\perp}\boldsymbol{\theta}$ denotes the component of the second order noise term orthogonal to $\bar{\boldsymbol{\theta}}$; since $\boldsymbol{\theta}$ is a unit vector, it has no error in the direction of itself, so we are interested in the error orthogonal to it (Fig. 6). Note that the first order error $\Delta_1\boldsymbol{\theta}$ in Eq. (64) is itself orthogonal to $\bar{\boldsymbol{\theta}}$. In Eq. (65), the matrix $\boldsymbol{T}$ is defined to be

$$\boldsymbol{T} \equiv \Delta_2\boldsymbol{M} - \Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^{-}\Delta_1\boldsymbol{M}. \tag{66}$$

From Eq. (64), we can show that the leading term of the covariance matrix of $\boldsymbol{\theta}$ has the following form [19].

$$V[\boldsymbol{\theta}] \equiv E[\Delta_1\boldsymbol{\theta}\Delta_1\boldsymbol{\theta}^{\top}] = \frac{\sigma^2}{N}\bar{\boldsymbol{M}}^{-}. \tag{67}$$

From this we observe:

- The covariance matrix $V[\boldsymbol{\theta}]$ is $O(\sigma^2)$.
- The right side of Eq. (19) agrees with the KCR lower bound.
- Eq. (67) does not contain the matrix $\boldsymbol{N}$.

Thus, we cannot change the value of Eq. (67) by adjusting the matrix $\boldsymbol{N}$. However, the root-mean-square (RMS) error of $\boldsymbol{\theta}$ is the sum of the covariance term and the bias term, and the bias term is also $O(\sigma^2)$ (the expectation of odd order noise terms is 0, so the first order bias is $E[\Delta_1\boldsymbol{\theta}] = \boldsymbol{0}$). Since the second order bias term contains the matrix $\boldsymbol{N}$, we can reduce it by adjusting $\boldsymbol{N}$. From Eq. (65), the second order bias has the following expression:

$$E[\Delta_2^{\perp}\boldsymbol{\theta}] = \bar{\boldsymbol{M}}^{-}\Big(\frac{(\bar{\boldsymbol{\theta}}, E[\boldsymbol{T}\bar{\boldsymbol{\theta}}])}{(\bar{\boldsymbol{\theta}}, \bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}})}\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}} - E[\boldsymbol{T}\bar{\boldsymbol{\theta}}]\Big). \tag{68}$$

4.4 Hyper-renormalization

Equation (68) implies that if we can choose an $\boldsymbol{N}$ such that

$$E[\boldsymbol{T}\bar{\boldsymbol{\theta}}] = c\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}} \tag{69}$$

for some constant c, we will have

$$E[\Delta_2^{\perp}\boldsymbol{\theta}] = \bar{\boldsymbol{M}}^{-}\Big(\frac{(\bar{\boldsymbol{\theta}}, c\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}})}{(\bar{\boldsymbol{\theta}}, \bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}})}\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}} - c\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}}\Big) = \boldsymbol{0}. \tag{70}$$

In other words, *the second order bias is completely eliminated*. In order to choose such an $\boldsymbol{N}$, we need to evaluate the expectation $E[\boldsymbol{T}\bar{\boldsymbol{\theta}}]$. After a detailed analysis (we omit the details), we can show that if we define the matrix $\bar{\boldsymbol{N}}$ by

$$\begin{aligned}\bar{\boldsymbol{N}} = &\frac{1}{N}\sum_{\alpha=1}^{N}\bar{W}_\alpha\Big(V_0[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[\bar{\boldsymbol{\xi}}_\alpha\boldsymbol{e}_\alpha^{\top}]\Big)\\ &-\frac{1}{N^2}\sum_{\alpha=1}^{N}\bar{W}_\alpha^2\Big((\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^{-}\bar{\boldsymbol{\xi}}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\\ &+2\mathcal{S}[V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{M}}^{-}\bar{\boldsymbol{\xi}}_\alpha\bar{\boldsymbol{\xi}}_\alpha^{\top}]\Big),\end{aligned} \tag{71}$$

then $E[\boldsymbol{T}\bar{\boldsymbol{\theta}}] = \sigma^2\bar{\boldsymbol{N}}\bar{\boldsymbol{\theta}}$ holds, where $\boldsymbol{e}_\alpha$ is a vector that depends on individual problems (the same vector as that in Eq. (48)), and $\mathcal{S}[\cdot]$ denotes symmetrization ($\mathcal{S}[\boldsymbol{A}] = (\boldsymbol{A} + \boldsymbol{A}^{\top})/2$). Since Eq. (71) contains the true values, they are replaced by computed values. This entails errors of $O(\sigma)$, but since the expectation of odd order noise terms is 0, Eq. (70) is $O(\sigma^4)$. Thus, we obtain the following procedure of *hyper-renormalization*:

1. Let $W_\alpha = 1$, $\alpha = 1, ..., N$, and $\boldsymbol{\theta}_0 = \boldsymbol{0}$.
2. Compute the following matrices $\boldsymbol{M}$ and $\boldsymbol{N}$:

$$\boldsymbol{M} = \frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top, \tag{72}$$

$$\begin{aligned}\boldsymbol{N} = &\frac{1}{N}\sum_{\alpha=1}^{N} W_\alpha \Big(V_0[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[\boldsymbol{\xi}_\alpha \boldsymbol{e}_\alpha^\top]\Big) \\ &-\frac{1}{N^2}\sum_{\alpha=1}^{N} W_\alpha^2 \Big((\boldsymbol{\xi}_\alpha, \boldsymbol{M}_{n-1}^- \boldsymbol{\xi}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \\ &+2\mathcal{S}[V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{M}_{n-1}^- \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top]\Big).\end{aligned} \tag{73}$$

   Here, $\boldsymbol{M}_{n-1}^-$ is the pseudoinverse of $\boldsymbol{M}$ with truncated rank $n-1$ (cf. Eq. (48).

3. Solve the generalized eigenvalue problem

$$\boldsymbol{M}\boldsymbol{\theta} = \lambda \boldsymbol{N}\boldsymbol{\theta}, \tag{74}$$

   and compute the unit eigenvector $\boldsymbol{\theta}$ for the smallest eigenvalue λ in absolute value.

4. If $\boldsymbol{\theta} \approx \boldsymbol{\theta}_0$ up to sign, return $\boldsymbol{\theta}$ and stop. Else, let

$$W_\alpha \leftarrow \frac{1}{(\boldsymbol{\theta}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{\theta})}, \quad \boldsymbol{\theta}_0 \leftarrow \boldsymbol{\theta}, \tag{75}$$

   and go back to Step 2.

It turns out that the initial solution with $W_\alpha = 1$ coincides with what is called *HyperLS* [23, 24, 39], which is derived to remove the bias up to second order error terms within the framework of algebraic methods without iterations[5]. We omit the details, but all the intermediate solutions $\boldsymbol{\theta}$ in the hyper-renormalization iterations are shown to be free of second oder bias. Thus, *hyper-renormalization is an iterative improvement of HyperLS*. The above procedure is for the case of single constraint in the form of Eq. (2) but can be straightforwardly extended to the case of multiple constraints (we omit the details).

4.5 Summary

We have seen that iterative reweight, renormalization, and hyper-renormalization do not minimize any cost function. In fact, *irrespective of their original motivations and derivations*, these are the methods for computing the "solution" $\boldsymbol{\theta}$ of

$$\boldsymbol{M}\boldsymbol{\theta} = \lambda \boldsymbol{N}\boldsymbol{\theta}. \tag{76}$$

Iterating generalized eigenvalue computation is merely a matter of convenience; any method that solves Eq. (76) can do. In this sense, (76) corresponds to the *esimating equation* for statistical estimation.

[5]The expression of Eq. (73) with $W_\alpha = 1$ lacks one term as compared with the corresponding expression of HyperLS, but the same solution is produced.

Table 1: Summary of the non-minimization approach.

initial solution		final solution
least squares	$\longrightarrow$	iterative reweight
Taubin	$\longrightarrow$	renormalization
HyperLS	$\longrightarrow$	hyper-renormalization

The matrix $\boldsymbol{M}$ on the left side determines the covariance of the resulting solution $\boldsymbol{\theta}$; if we choose the $\boldsymbol{M}$ in Eqs. (50), (58), and (72), the KCR lower bound is achieved up to $O(\sigma^4)$. The matrix $\boldsymbol{N}$ in Eq. (76) controls the bias. For iterative reweight, renormalization, and hyper-renormalization, it is chosen to be

$$\boldsymbol{N} = \begin{cases} \boldsymbol{I}, & \text{iterative reweight} \\ \dfrac{1}{N}\displaystyle\sum_{\alpha=1}^{N} W_\alpha V_0[\boldsymbol{\xi}_\alpha], & \text{renormalization} \\ \dfrac{1}{N}\displaystyle\sum_{\alpha=1}^{N} W_\alpha \Big(V_0[\boldsymbol{\xi}_\alpha] + 2\mathcal{S}[\boldsymbol{\xi}_\alpha \boldsymbol{e}_\alpha^\top]\Big) & \\ \quad -\dfrac{1}{N^2}\displaystyle\sum_{\alpha=1}^{N} W_\alpha^2 \Big(\cdots \Big). & \\ & \text{hyper-renormalization} \end{cases} \tag{77}$$

We have shown that the last choice (Eq. (73)) eliminates the bias up to $O(\sigma^4)$. If we iteratively solve Eq. (76) for the $\boldsymbol{N}$ in Eq. (77), the initial solution with $W_\alpha = 1$ corresponds to the least squares, the Taubin method [43], and HyperLS [23, 24, 39], respectively. In other words, Iterative reweight, renormalization, and hyper-renormalization can be viewed as iterative improvement of the solution of least squares, the Taubin method, and HyperLS, respectively (Table 1).

It can be shown that the matrix $\boldsymbol{N}$ of renormalization is positive semidefinite and has eigenvalue 0 while for hyper-renormalization $\boldsymbol{N}$ is neither positive nor negative (semi)definite: it has both positive and negative eigenvalues. Standard linear algebra routines for solving the generalized eigenvalue problem of Eq. (76) assume that $\boldsymbol{N}$ is positive definite, but the matrix $\boldsymbol{N}$ in Eq. (73) has both positive and negative eigenvalues. For renormalization, the matrix $\boldsymbol{N}$ is positive semidefinite, having eigenvalue 0. This, however, causes no trouble, because the problem can be rewritten as

$$\boldsymbol{N}\boldsymbol{\theta} = \frac{1}{\lambda}\boldsymbol{M}\boldsymbol{\theta}. \tag{78}$$

The matrix $\boldsymbol{M}$ is positive definite for noisy data, so we can use a standard routine to compute the eigenvector $\boldsymbol{\theta}$ for the eigenvalue $1/\lambda$ with the largest absolute value. If the matrix $\boldsymbol{M}$ happens to have eigenvalue 0, it indicates that the data are all exact, so

the unit eigenvector for the eigenvalue 0 is the exact solution.

The ML solution has bias of $O(\sigma^2)$ (cf. Sec. 3.9), while the bias of hyper-renormalization is $O(\sigma^4)$. For both, the covariance matrix of the solution agrees with the KCR lower bound up to $O(\sigma^4)$. In this sense, hyper-renormalization is more accurate than ML. In fact, according to experiments, the estimation error has roughly the following order:

LS > iterative reweight > Taubin > renormalization > ML > hyper-renormalization.

5. Examples

5.1 Evaluation of accuracy

Since the computed $\boldsymbol{\theta}$ and its true value $\bar{\boldsymbol{\theta}}$ are both unit vectors, we measure the discrepancy $\Delta\boldsymbol{\theta}$ between them by the orthogonal component to $\bar{\boldsymbol{\theta}}$ (Fig. 6),

$$\Delta^{\perp}\boldsymbol{\theta} = \boldsymbol{P}_{\bar{\boldsymbol{\theta}}}\boldsymbol{\theta}, \quad \boldsymbol{P}_{\bar{\boldsymbol{\theta}}} \equiv \boldsymbol{I} - \bar{\boldsymbol{\theta}}\bar{\boldsymbol{\theta}}^{\top}, \tag{79}$$

where $\boldsymbol{P}_{\bar{\boldsymbol{\theta}}}$ is the projection matrix along $\bar{\boldsymbol{\theta}}$. We generate M independent noise instances and evaluate the bias B and the RMS (root-mean-square) error D defined by

$$B = \Big\| \frac{1}{M}\sum_{a=1}^{M} \Delta^{\perp}\boldsymbol{\theta}^{(a)} \Big\|, \quad D = \sqrt{\frac{1}{M}\sum_{a=1}^{M} \|\Delta^{\perp}\boldsymbol{\theta}^{(a)}\|^2}, \tag{80}$$

where $\boldsymbol{\theta}^{(a)}$ is the solution in the ath trial. From the KCR lower bond in Eq. (19), the lower bound on the RMS error D is evaluated by

$$D \geq \frac{\sigma}{\sqrt{N}} \sqrt{\mathrm{tr}\Big(\frac{1}{N}\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_{\alpha}\bar{\boldsymbol{\xi}}_{\alpha}^{\top}}{(\bar{\boldsymbol{\theta}}, V_0[\boldsymbol{\xi}_{\alpha}]\bar{\boldsymbol{\theta}})} \Big)^{-}}, \tag{81}$$

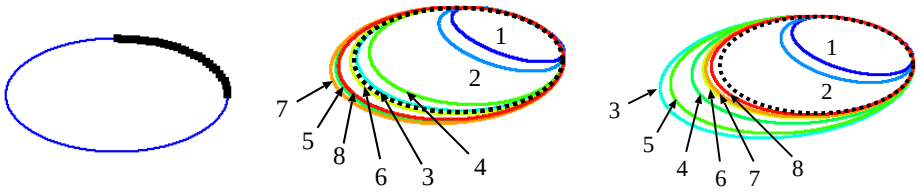
where tr denotes the matrix trace.

5.2 Ellipse fitting

We define 30 equidistant points on the ellipse shown in Fig. 7(a). The major and minor axis are set to 100 and 50 pixels, respectively. We add independent Gaussian noise of mean 0 and standard deviation σ (pixels) to the x and y coordinates of each point and fit an ellipse using: 1) LS, 2) iterative reweight, 3) the Taubin method, 4) renormalization, 5) HyperLS, 6) hyper-renormalization, 7) ML, and 8) hyperaccurate correction of ML. Figures 7(b), (c) show fitted ellipses for $\sigma = 0.5$ pixels; although the noise magnitude is the same, the resulting ellipses are different for different noise. The true shape is indicated by dotted lines. Iterative reweight, renormalization, and hyper-renormalization all converged after four iterations, while FNS for ML computation required nine iterations for Fig. 7(b) and eight iterations for Fig. 7(c). We can see that LS and iterative reweight have large bias, producing much smaller ellipses than the true shape. The closest ellipse is given by hyper-renormalization in Fig. 7(b) and by hyperaccurate correction of ML in Fig. 7(c). Since the solution is different for different noise, we need a statistical test for a fair comparison.

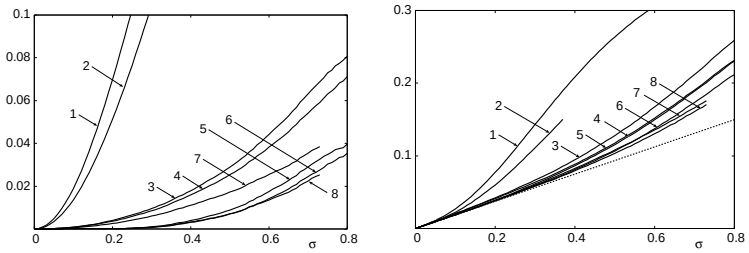
Figures 8(a), (b) plot the bias B and the RMS error D, respectively, defined in Eq. (80) over 10000 independent trials for each σ. The dotted line in Fig. 8(b) is the KCR lower bound of Eq. (81). The interrupted plots in Fig. 8 for iterative reweight, ML, and hyperaccurate correction of ML indicate that the iterations did not converge beyond that noise level. Our convergence criterion is $\|\boldsymbol{\theta}-\boldsymbol{\theta}_0\| < 10^{-6}$ for the current value $\boldsymbol{\theta}$ and the value $\boldsymbol{\theta}_0$ in the preceding iteration; their signs are adjusted before subtraction. If this criterion is not satisfied after 100 iterations, we stopped. For each σ, we regarded the iterations as not convergent if any among the 10000 trials did not converge. Figure 9 enlarges Fig. 8 for the small σ part. We can see from Fig. 8(a) that LS and iterative reweight have very large bias, in contrast to which the bias is very small for the Taubin method and renormalization. The bias of HyperLS and hyper-renormalization is still smaller and even smaller than ML. Since the leading covariance is common to iterative reweight, renormalization, and hyper-renormalization, the RMS error directly reflects the influence of the bias as shown in Fig. 8(b). Because hyper-renormalization does not have bias up to high order error terms, it has nearly the same accuracy as ML, or reprojection error minimization. A close examination of the small σ part (Fig. 9(b)) reveals that hyper-renormalization outperforms ML. The highest accuracy is achieved, although the difference is very small, by hyperaccurate correction of ML. However, it first requires the ML solution, and the FNS iterations for its computation may not converge above a certain noise level, as shown in Figs. 8. On the other hand, hyper-renormalization is very robust to noise. This is because the initial solution is HyperLS, which is itself highly accurate as seen from Figs. 8 and 9. For this reason, we conclude that it is the best method for practical computations.

Figure 10(a) is an edge image of a scene with a circular object. We fitted an ellipse to the 160 edge points indicated in red, using various methods. Figure 10(b) shows the fitted ellipses superimposed on the original image, where the occluded part is artificially composed for visual ease. In this case, iterative reweight converged after four iterations, and renormalization and hyper-renormalization converged after three iterations, while FNS for ML computation required six iterations. We can see that LS and iterative reweight produce much smaller ellipses than the true shape as in Fig. 7(b), (c). All other fits are very



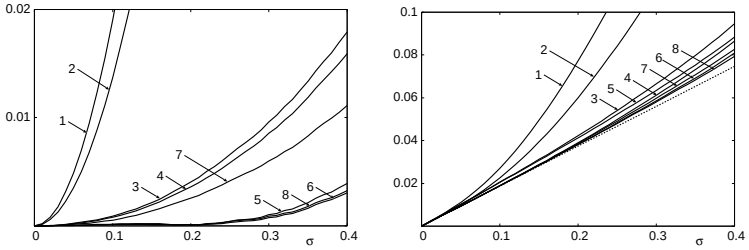
(a) (b) (c)

Figure 7: (a) Thirty points on an ellipse. (b), (c) Fitted ellipses ($\sigma = 0.5$ pixels). 1) LS, 2) iterative reweight, 3) Taubin, 4) renormalization, 5) HyperLS, 6) hyper-renormalization, 7) ML, 8) hyperaccurate correction of ML. The dotted lines indicate the true shape.



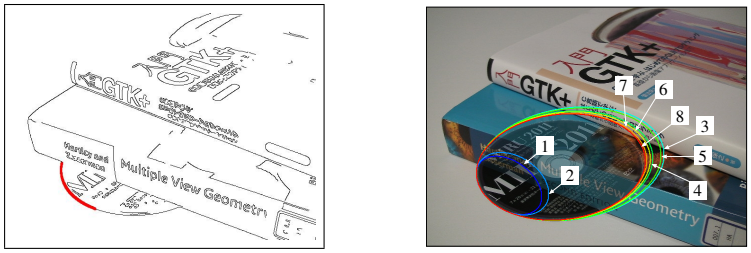
(a) (b)

Figure 8: The bias (a) and the RMS error (b) of the fitted ellipse for the standard deviation σ (pixels) of the noise added to the data in Fig. 7(a) over 10000 independent trials. 1) LS, 2) iterative reweight, 3) Taubin, 4) renormalization, 5) HyperLS, 6) hyper-renormalization, 7) ML, 8) hyperaccurate correction of ML. The dotted line in (b) indicates the KCR lower bound.



(a) (b)

Figure 9: (a) Enlargement of Fig. 8(a). (b) Enlargement of Fig. 8(b).



(a) (b)

Figure 10: (a) An edge image of a scene with a circular object. An ellipse is fitted to the 160 edge points indicated in red. (b) Fitted ellipses superimposed on the original image. The occluded part is artificially composed for visual ease. 1) LS, 2) iterative reweight, 3) Taubin, 4) renormalization, 5) HyperLS, 6) hyper-renormalization, 7) ML, 8) hyperaccurate correction of ML.

close to the true ellipse, and ML gives the best fit in this particular instance.

5.3 Fundamental matrix computation

Figure 11 shows simulated images of a curved grid surface viewed from two directions. The image size is 600×600 pixels, and the focal length is 600 pixels.

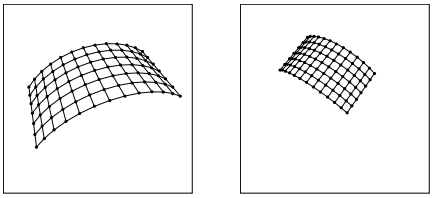


Figure 11: Simulated images of a curved grid surface viewed from two directions.

We add Gaussian noise of mean 0 and standard deviation σ (pixels) to the x and y coordinates of each grid point independently and compute the fundamental matrix $\boldsymbol{F}$. The fundamental matrix $\boldsymbol{F}$ has rank 2, so it is constrained to be $\det \boldsymbol{F} = 0$ [12]. As described in Sec. 3, this constraint can be imposed by the a posteriori correction, internal access, or external access approaches. Here, consider the a posteriori correction approach and compare the accuracy of various methods without considering the rank constraint.

Figures 12(a), (b) plot the bias B and the RMS error D, respectively, defined in Eq. (80) over 10000 independent trials for each σ. The dotted line in Fig. 12(b) is the KCR lower bound of Eq. (81). As we can see from Fig. 12(a), LS and iterative reweight have very large bias. As in the case of ellipse fitting, the leading covariance is common to iterative reweight, renormalization, and hyper-renormalization, and hence the RMS error reflects the influence of the bias as shown in Fig. 12(b). As seen from Fig. 12(a), ML has considerable bias, which is largely removed by the hyperaccurate correction, and hyper-renormalization directly computes solutions with as small bias. However, all methods except LS and iterative reweight nearly achieve the KCR lower bound, as shown in Fig. 12(b). Hence, the effect of bias reduction has little influence on the RMS error. As in the case of ellipse fitting, the best performance is obtained by hyper-renormalization and hyperaccurate correction of ML, although the difference from other methods is very small.

6. Concluding Remarks

6.1 Geometric estimation

Geometric estimation is different from statistical estimation. While statistical estimation is done by repeated sampling from a parameterized probability density (statistical noise), geometric estimation is done by observing one set of data that should ideally satisfy a parameterized implicit function (geometric model). Geometric estimation critically relies on the assumed statistical properties of noise and is classified into the minimization and non-minimization approaches. For whichever approach, there exists a theoretical accuracy limit called the KCR lower bound.

6.2 Minimization approach

Typical methods of the minimization approach are least squares (LS), maximum likelihood (ML), of which reprojection error minimization is a special case, and Sampson error minimization. The LS solution is easily computed by solving an eigenvalue problem but has large bias. The standard procedure for ML computation is to introduce auxiliary variables and search the high-dimensional joint parameter space, a typical application being bundle adjustment. Sampson error minimization is an approximation to ML, and the exact ML solution can be computed by iterating Sampson error minimization. However, the difference of ML from Sampson error minimization is very small, and they are practically equivalent. The accuracy of ML is further improved by hyperaccurate correction: the bias of ML is theoretically evaluated and is subtracted from the ML solution.

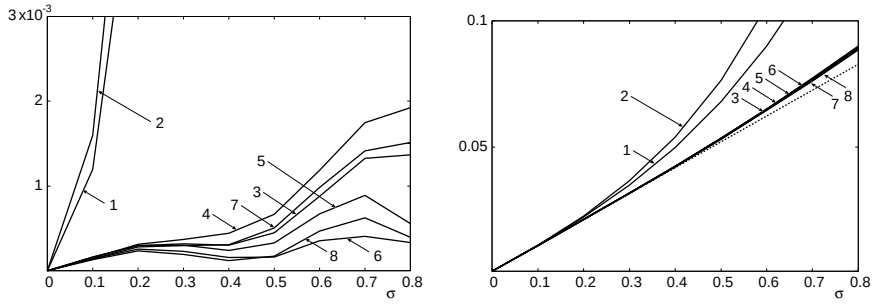
6.3 Non-minimization approach

Typical methods of the non-minimization approach are iterative reweight and renormalization. For both, the weight is iteratively updated so that the covariance matrix of solution achieves the KCR lower bound. We can modify the renormalization procedure so that the bias of the solution is eliminated up to high-order noise terms and derive hyper-renormalization. Iterative reweight, renormalization, and hyper-renormalization can be regarded as iterative improvement of LS, the Taubin method, and HyperLS, respectively. Viewed differently, iterative reweight, renormalization, and hyper-renormalization solve the generalized eigenvalue problem $\boldsymbol{M\theta} = \lambda \boldsymbol{N\theta}$, which correspond to the estimating equations for statistical estimation. The matrix $\boldsymbol{N}$ is not necessarily positive definite, but we can use standard library tools by solving $\boldsymbol{N\theta} = (1/\lambda)\boldsymbol{M\theta}$. Because renormalization and hyper-renormalization are respectively initialized by the Taubin method and HyperLS, which themselves are of high accuracy, the convergence is very fast; for many problems, the solution is obtained after three or four iterations.

6.4 Comparisons

According to numerical experiments, hyper-renormalization has higher accuracy than ML, which has been widely regarded as the most accurate method. The highest accuracy is achieved by hyperaccurate correction of ML, but the iterations for computing the ML solution may not converge in the presence of large noise. In contrast, hyper-renormalization is robust to noise. In this sense, hyper-renormalization is the best method for practical applications.

Acknowledgments Many parts of this paper are the results of the author's collaboration with Prasanna Ran-



(a) (b)

Figure 12: The bias (a) and the RMS error (b) of the computed fundamental matrix for the standard deviation σ (pixels) of the noise added to the data in Fig. 11 over 10000 independent trials. 1) LS, 2) iterative reweight, 3) Taubin, 4) renormalization, 5) HyperLS, 6) hyper-renormalization, 7) ML, 8) hyperaccurate correction of ML. The dotted line in (b) indicates the KCR lower bound.

garajan of Southern Methodist University, U.S.A., Ali Al-Sharadqah of the University of Mississippi, U.S.A. Nikolai Chernov of the University of Alabama at Birmingham, U.S.A., Yasuyuki Sugaya of Toyohashi University of Technology, Japan, and Chikara Matsunaga of FOR-A Co. Ltd., Japan. The author thanks Takayuki Okatani of Tohoku University, Japan, Mike Brooks and Wojciech Chojnacki of the University of Adelaide, Australia, Wolfgang Förstner of the University of Bonn, Germany, Peter Meer of Rutgers University, U.S.A., and Alexander Kukush of National Taras Shevchenko University of Kyiv, Ukraine, for helpful discussions. This work was supported in part by JSPS Grant-in-Aid for Challenging Exploratory Research (24650086).

References

[1] A. Al-Sharadqah and N. Chernov, A doubly optimal ellipse fit, *Comp. Stat. Data Anal.*, **56**-9 (2012-9), 2771–2781.

[2] S. Amari and M. Kawanabe, Information geometry of estimating functions in semiparametric statistical models, *Bernoulli*, **3** (1997-2), 29–54.

[3] P. J. Bickel, C. A. J. Klassen, Y. Ritov and J. A. Wellner, *Efficient and Adaptive Estimation for Semiparametric Models*, Johns Hopkins University Press, Baltimore, MD, U.S.A., 1994.

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

[5] N. Chernov and C. Lesort, Statistical efficiency of curve fitting algorithms, *Comp. Stat. Data Anal.*, **47**-4 (2004-11), 713–728.

[6] W. Chojnacki, M. J. Brooks and A. van den Hengel, Rationalising the renormalization method of Kanatani, *J. Math. Imaging Vis.*, **21**-11 (2001-2), 21–38.

[7] 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 (2000-11), 1294–1303.

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

[9] V. P. Godambe (Ed.), *Estimating Functions*, Oxford University Press, New York, U.S.A., 1991.

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

[11] R. Hartley and F. Kahl, Optimal algorithms in multiview geometry, *Proc. 8th Asian Conf. Comput. Vis.*, November 2007, Tokyo, Japan, Vol. 1, pp. 13–34.

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

[13] P. J. Huber, *Robust Statistics*, 2nd ed., Wiley, Hoboken, NJ, U.S.A., 2009.

[14] F. Kahl, S. Agarwal, M. K. Chandraker, D. Kriegman, and S. Belongie, Practical global optimization for multiview geometry, *Int. J. Comput. Vis.*, **79**-3 (2008-9), 271–284.

[15] K. Kanatani, Renormalization for unbiased estimation, *Proc. 4th Int. Conf. Comput. Vis.*, May 1993, Berlin, Germany, pp. 599–606.

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

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

[18] K. Kanatani, Ellipse fitting with hyperaccuracy, *IEICE Trans. Inf. & Syst.*, **E89-D**-10 (2006-10), 2653–2660.

[19] 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.

[20] K. Kanatani, A. Al-Sharadqah, N. Chernov, and Y. Sugaya, Renormalization returns: Hyper-renormalization and Its applications, *Proc. 12th Euro. Conf. Comput. Vis.*, October 2012, Florence, Italy, Vol. 3, pp. 358–398.

[21] K. Kanatani and H. Niitsuma, Optimal two-view planar triangulation *IPSJ Tran. Comput. Vis. Appl.*, **3** (2011-9), 67–79.

[22] K. Kanatani, N. Ohta and Y. Kanazawa, Optimal homography computation with a reliability measure, *IEICE Trans. Inf. & Syst.*, **E83-D**-7 (2000-7), 1369–1374.

[23] K. Kanatani and P. Rangarajan, Hyper least squares fitting of circles and ellipses, *Comput. Stat. Data Anal.*, **55**-6 (2011-6), 2197–2208.

[24] K. Kanatani, P. Rangarajan, Y. Sugaya and H. Niitsuma, HyperLS and its applications, *IPSJ Trans. Comput. Vis. Appl.*, **3** (2011-10), 80–94.

[25] K. Kanatani and Y. Sugaya, Extended FNS for constrained parameter estimation, *Proc Meeting Image Recog. Understand. 2007* July/August 2007, Hiroshima, Japan, pp. 219–226.

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

[27] K. Kanatani and Y. Sugaya, Compact algorithm for strictly ML ellipse fitting, *Proc. 19th Int. Conf. Patt. Recog.*, December 2008, Tampa, FL, U.S.A.

[28] K. Kanatani and Y. Sugaya, Compact fundamental matrix computation, *IPSJ Tran. Comput. Vis. Appl.*, **2** (2010-3), 59–70.

[29] K. Kanatani and Y. Sugaya, Unified computation of strict maximum likelihood for geometric fitting, *J. Math. Imaging Vis.*, **38**-1 (2010-9), 1–13.

[30] K. Kanatani and Y. Sugaya, Implementation and evaluation of bundle adjustment for 3-D reconstruction, *Pro. 17th Symp. Sensing via Imaging Information* June 2011, Yokohama, Japan, pp. IS4-02-1–IS4-02-8.

[31] Y. Leedan and P. Meer, Heteroscedastic regression in computer vision: Problems with bilinear constraint, *Int. J. Comput. Vis.*. **37**-2 (2000-6), 127-150.

[32] M. I. A. Lourakis and A. A. Argyros, SBA: A software package for generic sparse bundle adjustment, *ACM Trans. Math. Software*, **36**-1 (2009-3), 2:1–30.

[33] J. 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.

[34] J. Neyman and E. L. Scott, Consistent estimates based on partially consistent observations, *Econometrica*, **16**-1 (1948-1), 1–32.

[35] T. Okatani and K. Deguchi, Toward a statistically optimal method for estimating geometric relations from noisy data: Cases of linear relations, *Proc. IEEE Conf. Comput. Vis. Patt. Recog.*, June 2003, Madison, WI, U.S.A., Vol. 1, pp. 432–439.

[36] T. Okatani and K. Deguchi, On bias correction for geometric parameter estimation in computer vision, *Proc. IEEE Conf. Comput. Vis. Patt. Recog.*, June 2009, Miami Beach, FL, U.S.A., pp. 959-966.

[37] T. Okatani and K. Deguchi, Improving accuracy of geometric parameter estimation using projected score method, *Proc. Int. Conf. Comput. Vis.*, September/October 2009, Kyoto, Japan, pp. 1733-1740.

[38] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, 2nd ed., Cambridge University Press, Cambridge, U.K., 1992.

[39] P. Rangarajan and K. Kanatani, Improved algebraic methods for circle fitting, *Electronic J. Stat.*, **3** (2009-10), 1075–1082.

[40] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*, Wiley, New York, NY, U.S.A., 1987.

[41] P. D. Sampson, Fitting conic sections to "very scattered" data: An iterative refinement of the Bookstein algorithm, *Comput. Graphics Image Process.*, **18**-1 (1982-1), 97–108.

[42] P. Sturm and P. Gargallo, Conic fitting using the geometric distance, *Proc. 8th Asian Conf. Comput. Vis.*, November 2007, Tokyo, Japan, Vol. 2, pp.784–795.

[43] 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.

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