4th Int. Workshop Total Least Squares and Errors-in-Variables Modelling, August 2006, Leuven, Belgium.

Hyperaccuracy for Geometric Fitting

Kenichi Kanatani
Department of Computer Science, Okayama University, Okayama, Japan 700-8530
kanatani@suri.it.okayama-u.ac.jp

Abstract

A rigorous accuracy analysis is given to various techniques for estimating parameters of geometric models from noisy data. It is first pointed out that parameter estimation for computer vision applications is very different in nature from traditional statistical analysis and that a different mathematical framework is necessary in such a domain. After general theories on estimation and accuracy are given, typical existing techniques are selected, and their accuracy is evaluated up to higher order terms. This leads to a "hyperaccurate" method that outperforms existing methods.

1. Introduction

Modeling the geometric structure of images in a parametric form and estimating the parameters from observations are the first steps of many computer vision applications such as 3-D reconstruction and virtual reality generation. In the past, numerous optimization techniques have been proposed for such parameter estimation, but their accuracy is customarily tested using real and simulated images *a posteriori*. The purpose of this paper is to give a rigorous accuracy analysis of various estimation techniques *a priori*.

This sounds simple, because parameter estimation in the presence of noise is the main theme of statistics, so all one needs to do seems simply use the established results of statistics. We first point out that *this is not so* because parameter estimation for typical computer vision applications is very different in nature form traditional statistical analysis. We first discuss this in detail.

Next, we present a mathematical framework that specifically suits geometric computations frequently encountered in computer vision applications. This is in a sense "dual" to the standard paradigm found in the statistical literature.

After giving general theories on estimation and accuracy, we concentrate on problems for which the model equation can be transformed into a linear form via changes of variables. This type of problem covers most of the major computer vision applications. We select well known estimation techniques and analyze their accuracy up to higher order terms. This reveals why some methods known to be superior/inferior are really superior/inferior in quantitative terms. As a byproduct, our analysis leads to a "hyperaccurate" method that outperforms existing methods.

2. Geometric Fitting

2.1. Definition

We call the class of problems to be discussed in this paper *geometric fitting*: we fit a parameterized geometric model (a curve, a surface, or a relationship in high dimensions) expressed as an *implicit* equation in the form

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

to N data $\boldsymbol{x}_\alpha$, α = 1, ..., N, typically points in an image or point correspondences over multiple images [12]. The function $F(\boldsymbol{x}; \boldsymbol{u})$ is parameterized by vector $\boldsymbol{u}$. It may be a vector function if the model is defined by multiple equations. Each $\boldsymbol{x}_\alpha$ is assumed to be perturbed by independent noise from its true value $\bar{\boldsymbol{x}}_\alpha$ which strictly satisfies Eq. (1). From the parameter $\boldsymbol{u}$ of the fitted equation, one can discern the underlying geometric structure. A large class of computer vision problems fall into this category [12].

Though one can speak of noise and parameter estimation, the fact that this problem does *not* straightforwardly fit the traditional framework of statistics has not been widely recognized. The following are typical distinctions of geometric fitting as compared with the traditional parameter estimation problem:

- Unlike traditional statistics, there is no *explicit* model which explains observables in terms of deterministic mechanisms and random noise. All descriptions are *implicit*.
- No inputs or outputs exist. No such concepts exist as *causes* and *effects*, or *ordinates* and *abscissas*.
- The underlying data space is usually homogeneous and isotropic with no inherent coordinate system. Hence, the estimation process should be *invariant* to changes of the coordinate system with respect to which the data are described.
- Usually, the data are geometrically constrained. Typically, they are points on curves, surfaces, and hypersurfaces (e.g., unit vectors or matrices of determinant 0). Often, the parameters to be estimated are also similarly constrained. Hence, the *Gaussian distribution*,

the most fundamental noise modeling, does *not* exist in its strict sense in such constrained spaces.

We first discuss in detail why the traditional approach does not suit our intended applications.

2.2. Reduction to Statistical Estimation

It appears that the problem can be easily rewritten in the traditional form. The "observable" is the set of data $\boldsymbol{x}_\alpha$, which can be rearranged into a high dimensional vector $\boldsymbol{X} = \begin{pmatrix} \boldsymbol{x}_1^\top & \boldsymbol{x}_2^\top & \cdots & \boldsymbol{x}_N^\top \end{pmatrix}^\top$. Let $\boldsymbol{\varepsilon}_\alpha$ be the noise term in the datum $\boldsymbol{x}_\alpha$, and define the vector $\boldsymbol{E} = \begin{pmatrix} \boldsymbol{\varepsilon}_1^\top & \boldsymbol{\varepsilon}_2^\top & \cdots & \boldsymbol{\varepsilon}_N^\top \end{pmatrix}^\top$. Let $\bar{\boldsymbol{X}}$ be the true value of $\boldsymbol{X}$. The *statistical model* in the usual sense is

$$\boldsymbol{X} = \bar{\boldsymbol{X}} + \boldsymbol{E}. \tag{2}$$

The unknown $\bar{\boldsymbol{X}}$ needs to be estimated. Let $p(\boldsymbol{E})$ be the probability density of the noise vector $\boldsymbol{E}$. Our task is to estimate $\bar{\boldsymbol{X}}$ from $\boldsymbol{X}$, which we regard as sampled from $p(\boldsymbol{X} - \bar{\boldsymbol{X}})$. But the parameter $\boldsymbol{u}$, which we really want to estimate, is not contained in this model. How can we estimate it?

The truth is that the parameter $\boldsymbol{u}$ constrains the mutual relationships among the components of $\bar{\boldsymbol{X}}$. In fact, we would immediately obtain an optimal estimate $\bar{\boldsymbol{X}} = \boldsymbol{X}$ without considering such an implicit constraint.

In order to make the implicit constraint explicit, one needs to introducing *internal coordinates* $\boldsymbol{t}$ to solve Eq. (1) for $\boldsymbol{u}$ in the form

$$\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{t}; \boldsymbol{u}). \tag{3}$$

For example, if we want to fit a circle $(x-a)^2 + (y-b)^2 = r^2$, we rewrite this as $x = a + r\cos\theta$, $y = b + r\sin\theta$ by introducing the directional angle θ. However, this type of explicit representation is usually very difficult to obtain.

Suppose such an explicit representation is obtained. Substituting $\bar{\boldsymbol{x}}_1 = \boldsymbol{x}(\boldsymbol{t}_1, \boldsymbol{u})$, $\bar{\boldsymbol{x}}_2 = \boldsymbol{x}(\boldsymbol{t}_2, \boldsymbol{u})$, ..., $\bar{\boldsymbol{x}}_N = \boldsymbol{x}(\boldsymbol{t}_N, \boldsymbol{u})$, Eq. (2) now has the form

$$\boldsymbol{X} = \bar{\boldsymbol{X}}(\boldsymbol{t}_1, ..., \boldsymbol{t}_N; \boldsymbol{u}) + \boldsymbol{E}. \tag{4}$$

Our task is to estimate the parameters $\boldsymbol{t}_1$,..., $\boldsymbol{t}_N$ and $\boldsymbol{u}$ from $\boldsymbol{X}$.

2.3. Neyman-Scott Problem

Although the problem looks like a standard form, there is a big difference: we observe *only one* observable $\boldsymbol{X}$ for a "particular" set of parameters $\boldsymbol{t}_1$,..., $\boldsymbol{t}_N$ and $\boldsymbol{u}$. Namely, $\boldsymbol{X}$ is a *single* sample from $p(\boldsymbol{X} - \bar{\boldsymbol{X}}(\boldsymbol{t}_1, ..., \boldsymbol{t}_N; \boldsymbol{u}))$.

The tenet of statistical estimation is to observe *repeated* samples from a distribution, or *ensemble*, and infer its unknown parameters. The estimation becomes more accurate as more samples are drawn, thanks to the law of large numbers. Here, however, only one sample $\boldsymbol{X}$ is available.

What happens if we increase the data? If we observe another datum $\boldsymbol{x}_{N+1}$, the observable $\boldsymbol{X}$ becomes a higher dimensional vector, and Eq. (4) becomes a higher dimensional vector equation. Moreover, we have an additional unknown $\boldsymbol{t}_{N+1}$. This means that the resulting observable $\boldsymbol{X}$ is not "another" sample of the same distribution; it is *one* sample from a *new* distribution with a new set of parameters $\boldsymbol{t}_1$,..., $\boldsymbol{t}_{N+1}$ and $\boldsymbol{u}$. However large the number of data is, the number of observable is always 1.

This (seeming) anomaly was first pointed out by Neyman and Scott [21], and since then this problem has often been referred to as the *Neyman-Scott problem*. Even for a single observation, maximum likelihood estimation (ML) is possible. However, Neyman and Scott [21] pointed out that the estimated parameters do not necessarily converge to their true values as $N \to \infty$, indicating the (seeming) lack of "consistency", which is a characteristic of ML.

This is natural of course, because increasing the number of data does not mean increasing the number of *samples* from a distribution having *particular parameters*. Though $\boldsymbol{u}$ may be unchanged as N increases, we have as many parameters $\boldsymbol{t}_1$,..., $\boldsymbol{t}_N$ as the increased number of data. Due to this (seeming) anomaly, they are called *nuisance parameters*, while $\boldsymbol{u}$ the *structural parameter* or the *parameter of interest*.

2.4. Semiparametric Models

In spite of many attempts in the past, this anomaly has never been resolved, because it does not make sense to regard what is not standard statistical estimation as standard statistical estimation. It has been realized that the only way to fit the problem in the standard framework is to regard $\boldsymbol{t}_1$,..., $\boldsymbol{t}_N$ not as parameters but as *data* sampled from a *fixed* probability density $q(\boldsymbol{t}; \boldsymbol{v})$ with some unknown parameters $\boldsymbol{v}$ called *hyperparameters*.

The problem is now interpreted as follows. Given $\boldsymbol{u}$ and $\boldsymbol{v}$, the values $\boldsymbol{t}_1$,..., $\boldsymbol{t}_N$ are randomly sampled from $q(\boldsymbol{t}; \boldsymbol{v})$. Then, Eq. (3) defines the true values $\bar{\boldsymbol{x}}_1$, ..., $\bar{\boldsymbol{x}}_N$, to which random noise sampled from $p(\boldsymbol{E})$ is added. The task is to estimate both $\boldsymbol{u}$ and $\boldsymbol{v}$ by observing $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_N$. This approach is known as the *semiparametric model* [2, 4].

The standard procedure for such a problem goes like this. We first estimate or assume the density $q(\boldsymbol{t}; \boldsymbol{v})$ (the most difficult part), then *marginalize* the model over $q(\boldsymbol{t}; \boldsymbol{v})$, i.e., integrate out all $\boldsymbol{t}_1$, ..., $\boldsymbol{t}_N$ to obtain a likelihood function of $\boldsymbol{u}$ alone (not analytically easy), and finally search for the value $\boldsymbol{u}$ that maximizes it. In this case, the consistency as $N \to \infty$ is guaranteed under mild conditions.

This approach has been adopted in several computer vision problems where a large number of data are available. Ohta [22] showed that the semiparametric model yields a

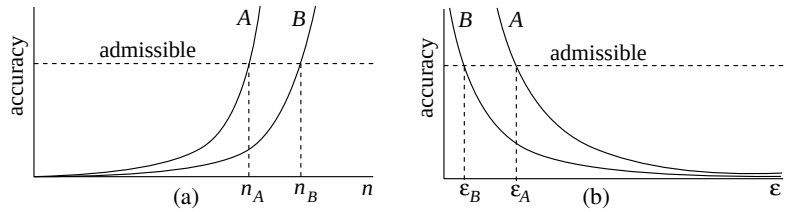

Figure 1. (a) For the standard statistical estimation, it is desired that the accuracy increases rapidly as $n \to \infty$ for the number n of observations, because admissible accuracy can be reached with a smaller number of observations. (b) For geometric fitting, it is desired that the accuracy increases rapidly as $\varepsilon \to 0$ for the noise level ε, because larger data uncertainty can be tolerated for admissible accuracy.

better result for 3-D interpretation of a dense optical flow field, and Okatani and Deguchi [23] demonstrated that for estimating 3-D shape and motion from a point cloud seen in multiple images, the semiparametric model can result in higher accuracy. In both cases, however, the procedure is very complicated, and the superior performance is obtained only when the number of data is extremely large.

2.5. Dual Approach of Kanatani

A natural question arises: why do we need to rewrite Eq. (1) in an explicit form by introducing internal coordinates $\boldsymbol{t}$? If Eq. (1) has a simple form, e.g., a polynomial, why do we need to convert it to a complicated (generally non-algebraic) form, if the conversion is possible at all. Indeed, it is known that a polynomial (or algebraic) equation does not have an algebraic explicit representation unless its "genus" is 0 (Clebsch theorem). Why can't we do estimation using Eq. (1) as is?

This might be answered as follows. Statisticians try to fit the problem in the standard framework because they are motivated to analyze *asymptotic* behavior of estimation as the number n of observations increases. In particular, the "consistency", i.e., the property that the computed estimates converge to their true value as $n \to \infty$, and the speed of convergence measured in $O((1/\sqrt{n})^k)$ are their major concern.

This is because an estimation method whose accuracy increases rapidly as $n \to \infty$ can attain admissible accuracy with a fewer number of observations (Fig. 1(a)). Such a method is desirable because most statistical applications are done in the presence of large noise (e.g., agriculture, medicine, economics, psychology, and census surveys), and hence one needs a large number of repeated observations to compensate for the noise, which entails a considerable cost in real situations.

To this, Kanatani [12, 14] countered, saying that the purpose of many computer vision applications is to estimate the underlying geometric structure as accurately as possible in the presence of small noise. In fact, the uncertainty introduced by image processing operations is usually around a few pixels or subpixels. He asserted that in such domains, it is more reasonable to evaluate the performance in the limit $\varepsilon \to 0$, because a method whose accuracy increases rapidly as $\varepsilon \to 0$ can tolerate larger uncertainty for admissible accuracy (Fig. 1(b)).

Table 1. Duality between traditional statistical estimation and geometric fitting [14].

statistical estimation	geometric fitting
data generating mechanism $\boldsymbol{x} \sim p(\boldsymbol{x}; \boldsymbol{\theta})$	geometric constraints $F(\boldsymbol{x}; \boldsymbol{u}) = 0$
CR lower bound $V_{\rm CR}[\hat{\theta}] = O(1/n)$	KCR lower bound $V_{\rm KCR}[\hat{\boldsymbol{u}}] = O(\varepsilon^2)$
ML is optimal in the limit $n \to \infty$	ML is optimal in the limit $\varepsilon \to 0$
Akaike's AIC AIC = $\cdots + O(1/n)$	geometric AIC G-AIC = $\cdots + O(\varepsilon^4)$
Rissanen's MDL MDL = $\cdots + O(1)$	geometric MDL G-MDL = $\cdots + O(\varepsilon^2)$

If our our interest is in the limit $\varepsilon \to 0$, we need not force Eq. (1) to conform to the traditional framework; we can build a mathematical theory of estimation directly from Eq. (1). Indeed, this is what has implicitly been done by many computer vision researchers for years without worrying much about orthodox theories in the statistical literature.

2.6. Duality of interpretation

Kanatani [12, 14] pushed this idea further in explicit terms and showed that resulting mathematical consequences have corresponding traditional results in a *dual* form, e.g., the KCR lower bound [5, 13] corresponds to the traditional Cramer-Rao (CR) lower bound, and the geometric AIC and the geometric MDL correspond, respectively, to Akaike's AIC [1] and Rissennen's MDL [25] (Table 1).

The correspondence is dual in the sense that small noise

expansions have the form $\cdots + O(\varepsilon^k)$ for geometric fitting, to which correspond traditional asymptotic expansions in the form $\cdots + O(1/\sqrt{n^k})$. Kanatani [12, 14] explained this, invoking the following thought experiment.

For geometric fitting, the image data may not be exact due to the uncertainty of image processing operations, but *they always have the same value however many times we observe them*, so the number n of observations is always 1, as pointed out earlier. Suppose, hypothetically, they change their values each time we observe them as if in quantum mechanics. Then, we would obtain n different values for n observations. If we take their sample mean, its standard deviation is $1/\sqrt{n}$ times that of individual observations. This means that repeating hypothetical observations n times effectively reduces the noise level ε to $\varepsilon/\sqrt{n}$. Thus, the behavior of estimation for $\varepsilon \to 0$ is mathematically equivalent to the asymptotic behavior for $n \to \infty$ of the number n of hypothetical observations (not the number N of "data").

In the following, we adopt this approach and analyze the accuracy of existing estimation techniques in the limit $\varepsilon \to 0$.

3. Parameter Estimation and Accuracy

3.1. Noise Description and Estimators

Our goal is to obtain a good estimate of the parameter $\boldsymbol{u}$ from observed data $\boldsymbol{x}_\alpha$. To do mathematical analysis, however, there is a serious obstacle arising from the fact that the data $\boldsymbol{x}_\alpha$ and the parameter $\boldsymbol{u}$ are usually constrained; they may be unit vectors or matrices of determinant 0, for instance. How can we define noise in the data and errors of the parameters? Evidently, direct vector calculus is not suitable. For example, if a unit vector is perturbed isotropically, the perturbed values are distributed around it over a unit sphere, but their average is not $\mathbf{0}$; it is "inside" the unit sphere.

A more serious problem is that noise distributions *cannot* be Gaussian, because Gaussian distributions with infinitely long tails can exist only in a Euclidean space. Since Gaussian distributions are the most fundamental of all distributions, how can we do mathematical analysis without it?

Several mathematical formulations have been proposed for probability distributions in a non-Euclidean space based on theories of Lie groups and invariant measures (e.g., Begelfor and Werman [3] and Pennec [24]), but the results are rather complicated. In our case, however, such complications are unnecessary, because we are focusing only on small noise effects in the dual framework. We can simply assume that noise concentrates within a small region around the true value. Hence, we can regard noise as effectively distributing over the *tangent space* to the domain at that point. Within this tangent space, the noise distribution

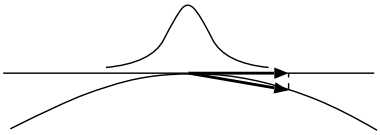

Figure 2. The displacement of a constrained variable is projected onto the tangent space, with which we identify the noise domain.

can be regarded as Gaussian; the discrepancy at the tail part is of higher order terms.

Accordingly, we define the *covariance matrix* of $\boldsymbol{x}_\alpha$ by

$$V[\boldsymbol{x}_\alpha] = E[\Big(\mathcal{P}_{\bar{\mathbf{x}}_\alpha}(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha)\Big)\Big(\mathcal{P}_{\bar{\mathbf{x}}_\alpha}(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha)\Big)^\top], \quad (5)$$

where $E[\,\cdot\,]$ denotes expectation over the noise distribution, and $\mathcal{P}_{\bar{\mathbf{x}}_\alpha}$ denotes projection onto the tangent space to the domain $\mathcal{X}$ of the data at $\bar{\boldsymbol{x}}_\alpha$ (Fig. 2).

The geometric fitting problem in the form of Eq. (1) is solved if a procedure is given for computing an estimate $\hat{\boldsymbol{u}}$ of $\boldsymbol{u}$ in terms of observed data $\boldsymbol{x}_\alpha$, which defines a function

$$\hat{\boldsymbol{u}} = \hat{\boldsymbol{u}}(\boldsymbol{x}_1, ..., \boldsymbol{x}_N), \quad (6)$$

called an *estimator* of $\boldsymbol{u}$. A natural requirement on this is that the true value should be obtained in the absence of noise:

$$\lim_{\varepsilon \to 0} \hat{\boldsymbol{u}} = \boldsymbol{u}. \quad (7)$$

Here, ε is the noise level, and $\boldsymbol{u}$ the true parameter value. Chernov and Lesort [5] called this condition *consistency* in the dual framework. In this paper, we consider only consistent estimators in this sense; confirming consistency is usually trivial.

If $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_N$ are random variables, so is $\hat{\boldsymbol{u}}$ as a function of them, so we can measure its accuracy by its covariance matrix. Here again, the parameter $\boldsymbol{u}$ may be constrained and its domain $\mathcal{U}$ may not be Euclidean. So, we identify the error of $\hat{\boldsymbol{u}}$ as belonging to the tangent space to $\mathcal{U}$ at the true value $\boldsymbol{u}$. To be specific, we define the covariance matrix $V[\hat{\boldsymbol{u}}]$ of $\hat{\boldsymbol{u}}$ by

$$V[\hat{\boldsymbol{u}}] = E[\Big(\mathcal{P}_{\mathbf{u}}(\hat{\boldsymbol{u}} - \boldsymbol{u})\Big)\Big(\mathcal{P}_{\mathbf{u}}(\hat{\boldsymbol{u}} - \boldsymbol{u})\Big)^\top], \quad (8)$$

where $\mathcal{P}_{\mathbf{u}}$ denotes projection onto the tangent space of the domain $\mathcal{U}$ at $\boldsymbol{u}$.

3.2. KCR Lower Bound

Kanatani [12, 15] proved that if each datum $\boldsymbol{x}_\alpha$ is an independent Gaussian random variable in the above-mentioned sense with mean $\bar{\boldsymbol{x}}_\alpha$ and covariance matrix

$V[\boldsymbol{x}_\alpha]$, the following inequality holds for an arbitrary unbiased estimator $\hat{\boldsymbol{u}}$ of $\boldsymbol{u}$ (see Appendix A for the proof):

$$V[\hat{\boldsymbol{u}}] \succ \left(\sum_{\alpha=1}^{N} \frac{(\mathcal{P}_{\mathbf{u}} \nabla_{\mathbf{u}} \bar{F}_\alpha)(\mathcal{P}_{\mathbf{u}} \nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{(\nabla_{\mathbf{x}} \bar{F}_\alpha, V[\boldsymbol{x}_\alpha] \nabla_{\mathbf{x}} \bar{F}_\alpha)} \right)^- . \tag{9}$$

Here, $\succ$ means that the left-hand side minus the right is positive semidefinite, and the superscript $-$ denotes pseudoinverse. The symbols $\nabla_{\mathbf{x}} \bar{F}_\alpha$ and $\nabla_{\mathbf{u}} \bar{F}_\alpha$ denote the gradient of the function $F(\boldsymbol{x}; \boldsymbol{u})$ in Eq. (1) with respect to $\boldsymbol{x}$ and $\boldsymbol{u}$, respectively, evaluated at $\boldsymbol{x} = \bar{\boldsymbol{x}}_\alpha$. Throughout this paper, we denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $(\boldsymbol{a}, \boldsymbol{b})$.

Chernov and Lesort [5] called the right-hand side of Eq. (9) the *KCR* (*Kanatani-Cramer-Rao*) *lower bound* and showed that it holds except for $O(\varepsilon^4)$ even if $\hat{\boldsymbol{u}}$ is not unbiased; it is sufficient that $\hat{\boldsymbol{u}}$ is "consistent" in the sense of Eq. (7).

If we worked in the traditional domain of statistics, we would obtain the corresponding *CR* (*Cramer-Rao*) *lower bound*. The statistical model is given by Eq. (4) with likelihood function $p(\boldsymbol{X} - \bar{\boldsymbol{X}}(\boldsymbol{t}_1, ..., \boldsymbol{t}_N; \boldsymbol{u}))$. We first compute second order derivatives of $\log p(\boldsymbol{X} - \bar{\boldsymbol{X}}(\boldsymbol{t}_1, ..., \boldsymbol{t}_N; \boldsymbol{u}))$ with respect to $\boldsymbol{t}_1$, ..., $\boldsymbol{t}_N$ and $\boldsymbol{u}$ (or multiply the first order derivatives) and define an $(mN + p) \times (mN + p)$ matrix, where m and p are the dimensions of the vectors $\boldsymbol{t}_\alpha$ and the vector $\boldsymbol{u}$, respectively. We then take expectation of this matrix with respect to $p(\boldsymbol{X} - \bar{\boldsymbol{X}}(\boldsymbol{t}_1, ..., \boldsymbol{t}_N; \boldsymbol{u}))$. The resulting matrix is called the *Fisher information matrix*. If we invert it and discard the nuisance parameters $\boldsymbol{t}_1$, ..., $\boldsymbol{t}_N$ by taking out only the $p \times p$ diagonal block corresponding to $\boldsymbol{u}$, we would obtain the CR lower bound on $\boldsymbol{u}$.

In most cases, this derivation process is almost intractable due to the difficulty of analytically inverting a matrix of a very large size. In contrast, the KCR lower bound in the form of Eq. (9) directly gives a bound without involving any "nuisance parameters". This is one of the most significant advantages of working in the dual framework of Kanatani [12, 15].

3.3. Minimization Schemes

It is a common strategy to define an estimator through minimization or maximization of some cost function, although this is not always necessary, as we will see later. Traditionally, the term "optimal" has been widely used to mean that something is minimized or maximized, and minimization or maximization has been simply called "optimization". Here, however, we reserve the term "optimal" for the strict sense that *nothing better can exists*.

A widely used method is what is called *least-squares estimation* (*LS*) (and by many other names such as *algebraic distance minimization*), minimizing

$$J = \sum_{\alpha=1}^{N} F(\boldsymbol{x}_\alpha; \boldsymbol{u})^2, \tag{10}$$

thereby implicitly defining an estimator $\hat{\boldsymbol{u}}(\boldsymbol{x}_1, ..., \boldsymbol{x}_N)$. It has been widely recognized that this estimator has low accuracy with large statistical bias. Another popular scheme is what is called *geometric distance minimization* (or by many other names), minimizing

$$J = \sum_{\alpha=1}^{N} \frac{F(\boldsymbol{x}_\alpha; \boldsymbol{u})^2}{\|\nabla_{\mathbf{x}} F_\alpha\|^2}. \tag{11}$$

Many other minimization schemes have been proposed in the past. All of them are designed so that $|F(\boldsymbol{x}_\alpha; \boldsymbol{u})|$ is as small as possible and at the same time the solution $\boldsymbol{u}$ has desirable properties. Kanatani [12] was the first to view the problem as *statistical estimation* for estimating the true data values $\bar{\boldsymbol{x}}_\alpha$ that strictly satisfy the constraint

$$F(\bar{\boldsymbol{x}}_\alpha; \boldsymbol{u}) = 0, \qquad \alpha = 1, ..., N, \tag{12}$$

using the knowledge of the data covariance matrices $V[\boldsymbol{x}_\alpha]$.

If we assume that the noise in each $\boldsymbol{x}_\alpha$ is independent Gaussian (in the tangent space) with mean $\mathbf{0}$ and covariance matrix $V[\boldsymbol{x}_\alpha]$, the likelihood of observing $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_N$ is

$$C \prod_{\alpha=1}^{N} e^{-(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha, V[\boldsymbol{x}_\alpha]^-(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha))/2}, \tag{13}$$

where C is a normalization constant. The true values $\bar{\boldsymbol{x}}_1$, ..., $\bar{\boldsymbol{x}}_N$ are constrained by Eq. (1). Maximizing Eq. (13) is equivalent to minimizing the negative of its logarithm (up to additive and multiplicative constants), called the (square) *Mahalanobis distance*,

$$J = \sum_{\alpha=1}^{N} (\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha, V[\boldsymbol{x}_\alpha]^-(\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha)), \tag{14}$$

subject to Eq. (12). Kanatani [12] called this scheme *maximum likelihood estimation* (*ML*) for geometric fitting.

The constraint of Eq. (12) can be eliminated by introducing Lagrange multipliers and ignoring higher order terms in the noise level, which can be justified in our dual framework. The resulting form is (see Appendix B for the derivation)

$$J = \sum_{\alpha=1}^{N} \frac{F(\boldsymbol{x}_\alpha; \boldsymbol{u})^2}{(\nabla_{\mathbf{x}} F_\alpha, V[\boldsymbol{x}_\alpha] \nabla_{\mathbf{x}} F_\alpha)}. \tag{15}$$

It can be shown that the covariance matrix $V[\hat{\boldsymbol{u}}]$ of the resulting estimator $\hat{\boldsymbol{u}}$ achieves the KCR lower bound except for $O(\varepsilon^4)$ [5, 12, 15] (see Appendix C for the proof). It is widely believed that this is the best method of all, aside from the semiparametric approach in the asymptotic limit $N \to \infty$.

3.4. Linearized Constraint Optimization

In the rest of this paper, we concentrate on a special subclass of geometric fitting problems in which Eq. (1) reduces to the linear form

$$(\boldsymbol{\xi}(\boldsymbol{x}), \boldsymbol{u}) = 0, \tag{16}$$

by changing variables $\boldsymbol{\xi} = \boldsymbol{\xi}(\boldsymbol{x})$. If the data $\boldsymbol{x}_\alpha$ are m-dimensional vectors and the unknown parameter $\boldsymbol{u}$ is a p-dimensional vector, $\boldsymbol{\xi}(\,\cdot\,)$ is a (generally nonlinear) embedding from $\mathcal{R}^m$ to $\mathcal{R}^p$. In order to remove scale indeterminacy, we normalize $\boldsymbol{u}$ to $\|\boldsymbol{u}\| = 1$.

The KCR lower bound for the linearized constraint has the form

$$V_{\rm KCR}[\hat{\boldsymbol{u}}] = \Big(\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\Big)^-, \tag{17}$$

where we write $\bar{\boldsymbol{\xi}}_\alpha = \boldsymbol{\xi}(\bar{\boldsymbol{x}}_\alpha)$. The covariance matrix $V[\boldsymbol{\xi}_\alpha]$ of $\boldsymbol{\xi}_\alpha = \boldsymbol{\xi}(\boldsymbol{x}_\alpha)$ is given, except for higher order terms in the noise level, in the form

$$V[\boldsymbol{\xi}_\alpha] = \nabla_{\mathbf{x}} \bar{\boldsymbol{\xi}}_\alpha^\top V[\boldsymbol{x}_\alpha] \nabla_{\mathbf{x}} \bar{\boldsymbol{\xi}}_\alpha, \tag{18}$$

where $\nabla_{\mathbf{x}} \bar{\boldsymbol{\xi}}_\alpha$ is the $m \times p$ Jacobian matrix

$$\nabla_{\mathbf{x}} \boldsymbol{\xi} = \begin{pmatrix} \partial \xi_1/\partial x_1 & \cdots & \partial \xi_p/\partial x_1 \\ \vdots & \ddots & \vdots \\ \partial \xi_1/\partial x_m & \cdots & \partial \xi_p/\partial x_m \end{pmatrix}. \tag{19}$$

evaluated at $\boldsymbol{x} = \bar{\boldsymbol{x}}_\alpha$. Note that in Eq. (17) we do not need the projection operator for the normalization constraint $\|\boldsymbol{u}\| = 1$, because $\bar{\boldsymbol{\xi}}_\alpha$ itself is orthogonal to $\boldsymbol{u}$ due to Eq. (16); for the moment, we assume that no other internal constraints exist.

This subclass of geometric fitting problems covers a wide range of computer vision applications. The following are typical examples:

Example 1 Suppose we want to fit a quadratic curve (circle, ellipse, parabola, hyperbola, or their degeneracy) to N points (x_α, y_α) in the plane. The constraint has the form

$$Ax^2 + 2Bxy + Cy^2 + 2(Dx + Ey) + F = 0. \tag{20}$$

If we define

$$\boldsymbol{\xi}(x, y) = (x^2 \;\; 2xy \;\; y^2 \;\; 2x \;\; 2y \;\; 1)^\top, \tag{21}$$

$$\boldsymbol{u} = (A \;\; B \;\; C \;\; D \;\; E \;\; F)^\top, \tag{22}$$

Eq. (20) is linearized in the form of Eq. (16). If independent Gaussian noise of mean 0 and standard deviation σ is added to each coordinates of (x_α, y_α), the covariance matrix $V[\boldsymbol{\xi}_\alpha]$ of the transformed $\boldsymbol{\xi}_\alpha$ has the form

$$V[\boldsymbol{\xi}_\alpha] = 4\sigma^2 \begin{pmatrix} \bar{x}_\alpha^2 & \bar{x}_\alpha \bar{y}_\alpha & 0 & \bar{x}_\alpha & 0 & 0 \\ \bar{x}_\alpha \bar{y}_\alpha & \bar{x}_\alpha^2 + \bar{y}_\alpha^2 & \bar{x}_\alpha \bar{y}_\alpha & \bar{y}_\alpha & \bar{x}_\alpha & 0 \\ 0 & \bar{x}_\alpha \bar{y}_\alpha & \bar{y}_\alpha^2 & 0 & \bar{y}_\alpha & 0 \\ \bar{x}_\alpha & \bar{y}_\alpha & 0 & 1 & 0 & 0 \\ 0 & \bar{x}_\alpha & \bar{y}_\alpha & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{23}$$

except for $O(\sigma^4)$, where $(\bar{x}_\alpha, \bar{y}_\alpha)$ is the true position of (x_α, y_α). □

Example 2 Suppose we have N corresponding points in two images of the same scene viewed from different positions. If point (x, y) in the first image corresponds to (x', y') in the second, they should satisfy the following *epipolar equation* [10]:

$$\Big(\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \boldsymbol{F} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix}\Big) = 0. \tag{24}$$

Here, $\boldsymbol{F}$ is a matrix of rank 2, called the *fundamental matrix*, that depends only on the intrinsic parameters of the two cameras that took the two images and their relative 3-D positions (i.e., independent of the scene and the location of the identified points) [10]. If we define

$$\boldsymbol{\xi}(x, y, x', y') = (xx' \;\; xy' \;\; x \;\; yx' \;\; yy' \;\; y \;\; x' \;\; y' \;\; 1)^\top, \tag{25}$$

$$\boldsymbol{u} = (F_{11} \;\; F_{12} \;\; F_{13} \;\; F_{21} \;\; F_{22} \;\; F_{23} \;\; F_{31} \;\; F_{32} \;\; F_{33})^\top, \tag{26}$$

Eq. (24) is linearized in the form of Eq. (16). If independent Gaussian noise of mean 0 and standard deviation σ is added to each coordinates of the corresponding points (x_α, y_α) and (x'_α, y'_α), the covariance matrix $V[\boldsymbol{\xi}_\alpha]$ of the transformed $\boldsymbol{\xi}_\alpha$ has the form

$$V[\boldsymbol{\xi}_\alpha] = \sigma^2 \times \begin{pmatrix} \bar{x}_\alpha^2 + \bar{x}_\alpha'^2 & \bar{x}'_\alpha \bar{y}'_\alpha & \bar{x}'_\alpha & \bar{x}_\alpha \bar{y}_\alpha & 0 & 0 & \bar{x}_\alpha & 0 & 0 \\ \bar{x}'_\alpha \bar{y}'_\alpha & \bar{x}_\alpha^2 + \bar{y}_\alpha'^2 & \bar{y}'_\alpha & 0 & \bar{x}_\alpha \bar{y}_\alpha & 0 & 0 & \bar{x}_\alpha & 0 \\ \bar{x}'_\alpha & \bar{y}'_\alpha & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \bar{x}_\alpha \bar{y}_\alpha & 0 & 0 & \bar{y}_\alpha^2 + \bar{x}_\alpha'^2 & \bar{x}'_\alpha \bar{y}'_\alpha & \bar{x}'_\alpha & \bar{y}_\alpha & 0 & 0 \\ 0 & \bar{x}_\alpha \bar{y}_\alpha & 0 & \bar{x}'_\alpha \bar{y}'_\alpha & \bar{y}_\alpha^2 + \bar{y}_\alpha'^2 & \bar{y}'_\alpha & 0 & \bar{y}_\alpha & 0 \\ 0 & 0 & 0 & \bar{x}'_\alpha & \bar{y}'_\alpha & 1 & 0 & 0 & 0 \\ \bar{x}_\alpha & 0 & 0 & \bar{y}_\alpha & 0 & 0 & 1 & 0 & 0 \\ 0 & \bar{x}_\alpha & 0 & 0 & \bar{y}_\alpha & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{27}$$

except for $O(\sigma^4)$, where $(\bar{x}_\alpha, \bar{y}_\alpha)$ and $(\bar{x}'_\alpha, \bar{y}'_\alpha)$, are the true positions of (x_α, y_α) and (x'_α, y'_α), respectively. The fundamental matrix has, aside from scale normalization, the constraint that its determinant is 0. If we take this constraint into consideration, the KCR lower bound of Eq (17) involves the corresponding projection operation [17]. □

As we can see from Eqs. (23) and (27), the covariance matrix $V[\boldsymbol{\xi}_\alpha]$ is often factored into the form

$$V[\boldsymbol{\xi}_\alpha] = \varepsilon^2 V_0[\boldsymbol{\xi}_\alpha], \tag{28}$$

where ε is a constant that characterizes the noise and $V_0[\boldsymbol{\xi}_\alpha]$ is a matrix that depends only on the true data values. Hereafter, we assume this form and call ε the *noise level* and $V_0[\boldsymbol{\xi}_\alpha]$ the *normalized covariance matrix*. In the actual computation, the true data values are approximated by their observed values.

4. Accuracy of Parameter Estimation

4.1. Least-Squares Method

For the linearized constraint of Eq. (16), minimization of Eq. (10) reduces to minimization of

$$J = \sum_{\alpha=1}^{N} (\boldsymbol{\xi}_\alpha, \boldsymbol{u})^2 = \sum_{\alpha=1}^{N} (\boldsymbol{u}, \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top \boldsymbol{u}) = (\boldsymbol{u}, \boldsymbol{M}_0 \boldsymbol{u}), \tag{29}$$

where

$$\boldsymbol{M}_0 = \sum_{\alpha=1}^{N} \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top. \tag{30}$$

This is a symmetric matrix (generally positive definite), so the quadratic form $(\boldsymbol{u}, \boldsymbol{M}_0 \boldsymbol{u})$ is minimized by the unit eigenvector for the smallest eigenvalue of $\boldsymbol{M}_0$.

To do error analysis, we write

$$\boldsymbol{M}_0 \hat{\boldsymbol{u}} = \lambda \hat{\boldsymbol{u}}, \tag{31}$$

into which we substitute $\boldsymbol{\xi}_\alpha = \bar{\boldsymbol{\xi}}_\alpha + \Delta\boldsymbol{\xi}_\alpha$ and $\hat{\boldsymbol{u}} = \boldsymbol{u} + \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{u} + \cdots$, where Δ_1 and Δ_2 denote perturbations corresponding to the first and the second orders in $\Delta\boldsymbol{\xi}_\alpha$, respectively. We have

$$\begin{aligned}&(\bar{\boldsymbol{M}}_0 + \Delta_1\boldsymbol{M}_0 + \Delta_2\boldsymbol{M}_0)(\boldsymbol{u} + \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{u} + \cdots)\\ &= (\Delta_1\lambda + \Delta_2\lambda + \cdots)(\boldsymbol{u} + \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{u} + \cdots),\end{aligned} \tag{32}$$

where $\bar{\boldsymbol{M}}_0$ is the value of $\boldsymbol{M}_0$ obtained by replacing $\boldsymbol{\xi}_\alpha$ in Eq. (31) by their true values $\bar{\boldsymbol{\xi}}_\alpha$, and

$$\Delta_1\boldsymbol{M}_0 = \sum_{\alpha=1}^{N} (\bar{\boldsymbol{\xi}}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top + \Delta\boldsymbol{\xi}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top), \tag{33}$$

$$\Delta_2\boldsymbol{M}_0 = \sum_{\alpha=1}^{N} \Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top. \tag{34}$$

We also expand the eigenvalue λ in Eq. (31) into $\Delta_1\lambda + \Delta_2\lambda + \cdots$. Since $\lambda = 0$ in the absence of noise, its 0th order term does not exist.

Equating first and second order terms on both sides of Eq. (32), we obtain

$$\bar{\boldsymbol{M}}_0 \Delta_1\boldsymbol{u} + \Delta_1\boldsymbol{M}_0 \boldsymbol{u} = \Delta_1\lambda \boldsymbol{u}, \tag{35}$$

$$\bar{\boldsymbol{M}}_0 \Delta_2\boldsymbol{u} + \Delta_1\boldsymbol{M}_0 \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{M}_0 \boldsymbol{u} = \Delta_2\lambda \boldsymbol{u}. \tag{36}$$

Computing the inner product with $\boldsymbol{u}$ on both sides of Eq. (35) and noting that $(\boldsymbol{u}, \bar{\boldsymbol{M}}_0\boldsymbol{u})$ and $(\boldsymbol{u}, \Delta\bar{\boldsymbol{M}}_0\boldsymbol{u})$ identically vanish, we see that $\Delta_1\lambda = 0$. Multiplying $\bar{\boldsymbol{M}}_0^-$ on both sides of Eq. (35) and noting that $\bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{M}}_0 = \boldsymbol{P}_{\mathbf{u}}$ ($\equiv \boldsymbol{I} - \boldsymbol{u}\boldsymbol{u}^\top$, the projection matrix onto the hyperplane orthogonal to $\boldsymbol{u}$) and $\Delta_1\boldsymbol{u}$ is orthogonal to $\boldsymbol{u}$ to a first approximation (because $\|\boldsymbol{u}\| = 1$), we conclude that

$$\Delta_1\boldsymbol{u} = -\bar{\boldsymbol{M}}_0^- \Delta_1\boldsymbol{M}_0 \boldsymbol{u}. \tag{37}$$

Evidently, we have $E[\Delta_1\boldsymbol{u}] = \boldsymbol{0}$. The covariance matrix it defines is

$$\begin{aligned}V[\hat{\boldsymbol{u}}] &= E[\Delta_1\boldsymbol{u}\Delta_1\boldsymbol{u}^\top]\\ &= \bar{\boldsymbol{M}}_0^- E[(\Delta_1\boldsymbol{M}_0\boldsymbol{u})(\Delta_1\boldsymbol{M}_0\boldsymbol{u})^\top]\bar{\boldsymbol{M}}_0^-\\ &= \bar{\boldsymbol{M}}_0^- E[\sum_{\alpha=1}^{N}(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha \sum_{\beta=1}^{N}(\Delta\boldsymbol{\xi}_\beta, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\beta^\top]\bar{\boldsymbol{M}}_0^-\\ &= \bar{\boldsymbol{M}}_0^- \sum_{\alpha\beta=1}^{N} (\boldsymbol{u}, E[\Delta\boldsymbol{\xi}_\alpha\Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha\bar{\boldsymbol{\xi}}_\beta^\top \bar{\boldsymbol{M}}_0^-\\ &= \varepsilon^2 \bar{\boldsymbol{M}}_0^- \Big(\sum_{\alpha=1}^{N} (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha\bar{\boldsymbol{\xi}}_\beta^\top\Big)\bar{\boldsymbol{M}}_0^-,\end{aligned} \tag{38}$$

where we use the identity $E[\Delta\boldsymbol{\xi}_\alpha\Delta\boldsymbol{\xi}_\beta^\top] = \varepsilon^2\delta_{\alpha\beta}V_0[\boldsymbol{\xi}_\alpha]$ implied by our assumption about the noise ($\delta_{\alpha\beta}$ is the Kronecker delta, taking 1 for $\alpha = \beta$ and 0 otherwise).

Multiplying $\bar{\boldsymbol{M}}_0^-$ on both sides of Eq. (36) and solving for $\bar{\boldsymbol{M}}_0^-\bar{\boldsymbol{M}}_0\Delta_2\boldsymbol{u}$ ($\equiv \boldsymbol{P}_{\mathbf{u}}\Delta_2\boldsymbol{u}$), we obtain

$$\begin{aligned}&\Delta_2\boldsymbol{u}^\perp\\ &= -\bar{\boldsymbol{M}}_0^- \Delta_1\boldsymbol{M}_0\Delta_1\boldsymbol{u} - \bar{\boldsymbol{M}}_0^- \Delta_2\boldsymbol{M}_0\boldsymbol{u}\\ &= \bar{\boldsymbol{M}}_0^- \Delta_1\boldsymbol{M}_0\bar{\boldsymbol{M}}_0^- \Delta_1\boldsymbol{M}_0\boldsymbol{u} - \bar{\boldsymbol{M}}_0^- \Delta_2\boldsymbol{M}_0\boldsymbol{u},\end{aligned} \tag{39}$$

where $\Delta_2\boldsymbol{u}^\perp$ ($\equiv \boldsymbol{P}_{\mathbf{u}}\Delta_2\boldsymbol{u}$) is the component of $\Delta_2\boldsymbol{u}$ orthogonal to $\boldsymbol{u}$. The parallel component $\Delta_2\boldsymbol{u}^\parallel$ can also be computed, but it is not important, since it arises solely for enforcing the normalization constraint $\|\hat{\boldsymbol{u}}\|^2 = 1$ (Fig. 3). Hence, we can measure the accuracy only by examining the orthogonal component, as discussed in Section 3.1.

If we note that

$$\begin{aligned}&E[\Delta_1\boldsymbol{M}_0\bar{\boldsymbol{M}}_0^- \Delta_1\boldsymbol{M}_0\boldsymbol{u}]\\ &= E[\sum_{\alpha=1}^{N}(\bar{\boldsymbol{\xi}}_\alpha\Delta\boldsymbol{\xi}_\alpha^\top + \Delta\boldsymbol{\xi}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top)\bar{\boldsymbol{M}}_0^- \sum_{\beta=1}^{N}(\Delta\boldsymbol{\xi}_\beta, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\beta]\\ &= \sum_{\alpha,\beta=1}^{N} (\boldsymbol{u}, E[\Delta\boldsymbol{\xi}_\beta\Delta\boldsymbol{\xi}_\alpha^\top]\bar{\boldsymbol{M}}_0^-\bar{\boldsymbol{\xi}}_\beta)\bar{\boldsymbol{\xi}}_\alpha\\ &\quad + \sum_{\alpha,\beta=1}^{N} (\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}_0^-\bar{\boldsymbol{\xi}}_\beta)E[\Delta\boldsymbol{\xi}_\alpha\Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u}\end{aligned}$$

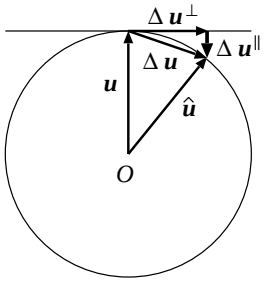

Figure 3. The orthogonal error component $\Delta u^{\perp}$ and the parallel error component $\Delta u^{\parallel}$ of an estimate $\hat{u}$ of u. The accuracy can be measured by the orthogonal component $\Delta u^{\perp}$.

$$= \varepsilon^2 \sum_{\alpha=1}^{N} (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) \bar{\boldsymbol{\xi}}_\alpha + \varepsilon^2 \sum_{\alpha=1}^{N} (\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}, \quad (40)$$

$$E[\Delta_2 \boldsymbol{M}_0 \boldsymbol{u}] = \sum_{\alpha=1}^{N} E[\Delta \boldsymbol{\xi}_\alpha \Delta \boldsymbol{\xi}_\alpha^\top] \boldsymbol{u} = \varepsilon^2 \sum_{\alpha=1}^{N} V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u} = \varepsilon^2 \boldsymbol{N}_0 \boldsymbol{u}, \quad (41)$$

where we define

$$\boldsymbol{N}_0 = \sum_{\alpha=1}^{N} V_0[\boldsymbol{\xi}_\alpha], \quad (42)$$

the expectation of $\Delta_2 \boldsymbol{u}^{\perp}$ is given by

$$E[\Delta_2 \boldsymbol{u}^{\perp}] = \varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^{N} (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) \bar{\boldsymbol{\xi}}_\alpha + \varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^{N} (\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u} - \varepsilon^2 \bar{\boldsymbol{M}}_0^- \boldsymbol{N}_0 \boldsymbol{u}. \quad (43)$$

4.2. Taubin Method

The method due to Taubin[1] [26] is to minimize, instead of Eq. (29),

$$J = \frac{\sum_{\alpha=1}^{N} (\boldsymbol{\xi}_\alpha, \boldsymbol{u})^2}{\sum_{\alpha=1}^{N} (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u})} = \frac{(\boldsymbol{u}, \boldsymbol{M}_0 \boldsymbol{u})}{(\boldsymbol{u}, \boldsymbol{N}_0 \boldsymbol{u})}. \quad (44)$$

[1]Taubin [26] studied curve fitting, which he analyzed purely from a geometric point of view without using statistical terms such as means and covariance matrices. What is shown here is a modification of his method in the present framework.

This is a Rayleigh ratio, so it is minimized by the eigenvector of the generalized eigenvalue problem

$$\boldsymbol{M}_0 \hat{\boldsymbol{u}} = \lambda \boldsymbol{N}_0 \hat{\boldsymbol{u}}, \quad (45)$$

for the smallest eigenvalue. The matrix $\boldsymbol{N}_0$ may be singular, but we can solve Eq. (45) by reducing the number of parameters as prescribed by Chojnacki, et al. [8, 9] (see Appendix D for the procedure).

As in the case of LS, we expand Eq. (45) in the form

$$(\bar{\boldsymbol{M}}_0 + \Delta_1 \boldsymbol{M}_0 + \Delta_2 \boldsymbol{M}_0)(\boldsymbol{u} + \Delta_1 \boldsymbol{u} + \Delta_2 \boldsymbol{u} + \cdots) = (\Delta_1 \lambda + \Delta_2 \lambda + \cdots) \boldsymbol{N}_0 (\boldsymbol{u} + \Delta_1 \boldsymbol{u} + \Delta_2 \boldsymbol{u} + \cdots), \quad (46)$$

and equate first and second order terms on both sides. We obtain

$$\bar{\boldsymbol{M}}_0 \Delta_1 \boldsymbol{u} + \Delta_1 \boldsymbol{M}_0 \boldsymbol{u} = \Delta_1 \lambda \boldsymbol{N}_0 \boldsymbol{u}, \quad (47)$$

$$\bar{\boldsymbol{M}}_0 \Delta_2 \boldsymbol{u} + \Delta_1 \boldsymbol{M}_0 \Delta_1 \boldsymbol{u} + \Delta_2 \boldsymbol{M}_0 \boldsymbol{u} = \Delta_2 \lambda \boldsymbol{N}_0 \boldsymbol{u}. \quad (48)$$

Computing the inner product with $\boldsymbol{u}$ on both sides of Eq. (47), we again find that $\Delta_1 \lambda = 0$. So, the first order error $\Delta_1 \boldsymbol{u}$ is also given by Eq. (37) and hence the covariance matrix $V[\hat{\boldsymbol{u}}]$ by Eq. (38). In other words, LS and the Taubin method have the same accuracy to a first approximation.

However, the Taubin method is known to be substantially better than LS. So, the difference should be second-order effects. Multiplying $\bar{\boldsymbol{M}}_0^-$ on both sides of Eq. (48) and solving for $\Delta_2 \boldsymbol{u}^{\perp}$ ($\equiv \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{M}}_0^- \Delta_2 \boldsymbol{u}$), we obtain

$$\Delta_2 \boldsymbol{u}^{\perp} = -\bar{\boldsymbol{M}}_0^- \Delta_1 \boldsymbol{M}_0 \Delta_1 \boldsymbol{u} - \bar{\boldsymbol{M}}_0^- \Delta_2 \boldsymbol{M}_0 \boldsymbol{u} - \Delta_2 \lambda \bar{\boldsymbol{M}}_0 \boldsymbol{N} \boldsymbol{u} = \bar{\boldsymbol{M}}_0^- \Delta_1 \boldsymbol{M}_0 \bar{\boldsymbol{M}}_0^- \Delta_1 \boldsymbol{M}_0 \boldsymbol{u} - \bar{\boldsymbol{M}}_0^- \Delta_2 \boldsymbol{M}_0 \boldsymbol{u} - \Delta_2 \lambda \bar{\boldsymbol{M}}_0 \boldsymbol{N} \boldsymbol{u}. \quad (49)$$

Comparing this with Eq. (39), we find that an extra term, $-\Delta_2 \lambda \bar{\boldsymbol{M}}_0^- \boldsymbol{N} \boldsymbol{u}$, is added. We now evaluate its expectation.

Computing the inner product with $\boldsymbol{u}$ on both sides of Eq. (48), and noting that $(\boldsymbol{u}, \boldsymbol{M}_0 \Delta_2 \boldsymbol{u})$ and $(\boldsymbol{u}, \Delta_1 \boldsymbol{M}_0 \Delta_1 \boldsymbol{u})$ identically vanish, we obtain

$$\Delta_2 \lambda = \frac{(\boldsymbol{u}, \Delta_2 \boldsymbol{M}_0 \boldsymbol{u})}{(\boldsymbol{u}, \boldsymbol{N}_0 \boldsymbol{u})}. \quad (50)$$

Its expectation is

$$E[\Delta_2 \lambda] = \frac{(\boldsymbol{u}, E[\Delta_2 \boldsymbol{M}_0 \boldsymbol{u}])}{(\boldsymbol{u}, \boldsymbol{N}_0 \boldsymbol{u})} = \varepsilon^2, \quad (51)$$

where we have used Eq. (41). Hence, the expectation of the last term in Eq. (49) cancel the last term of Eq. (43), resulting in

$$E[\Delta_2 \boldsymbol{u}^{\perp}] = \varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^{N} (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) \bar{\boldsymbol{\xi}}_\alpha + \varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^{N} (\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}. \quad (52)$$

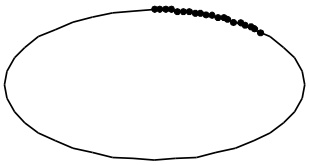

Figure 4. 20 points on an ellipse.

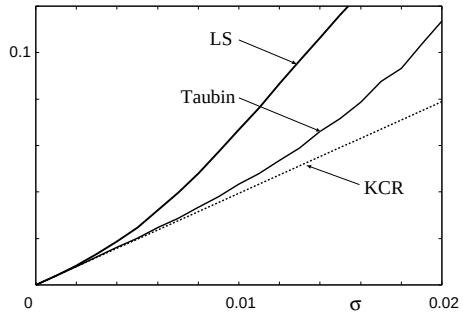

Figure 5. Noise level vs. RMS error for the ellipse data in Fig. 4: LS (thick solid line), Taubin (thin solid line), and KCR lower bound (dotted line).

In other words, the second order bias $-\varepsilon^2 \boldsymbol{M}_0^- \boldsymbol{N}_0 \boldsymbol{u}$ of LS is eliminated by the introduction of $\boldsymbol{N}_0$ on the right-hand side of Eq. (45). We conclude that this is the cause of the improved accuracy of the Taubin method as compared with LS.

Example 3 Figure 4 shows N = 20 points $(\bar{x}_\alpha, \bar{y}_\alpha)$ taken on ellipse

$$\frac{x^2}{50^2} + \frac{y^2}{100^2} = 1, \tag{53}$$

with equal intervals. From them, we generated data points (x_α, y_α) by adding Gaussian noise of mean 0 and standard deviation σ to the x and y coordinates independently. Then, we fitted an ellipse by LS and the Taubin method.

Figure 5 plots for different σ the fitting error evaluated by the following root mean square over 10,000 independent trials:

$$D = \sqrt{\frac{1}{10000} \sum_{a=1}^{10000} \|\boldsymbol{P}_{\mathbf{u}} \hat{\boldsymbol{u}}^{(a)}\|^2}. \tag{54}$$

Here, $\hat{\boldsymbol{u}}^{(a)}$ is the ath value of $\hat{\boldsymbol{u}}$. The thick and thin line are for LS and the Taubin method, respectively. The dotted line is the root mean square given by the corresponding KCR lower (tr denotes the trace):

$$D_{\mathrm{KCR}} = 2\sigma \sqrt{\mathrm{tr}\Big(\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\Big)^-}. \tag{55}$$

As we can see, the LS solution is of very low accuracy, while the Taubin solution is fairly accurate. The plots for LS and Taubin should have, at σ = 0, the same slope distinct from that of the KCR lower bound, as far as the first order error $\Delta_1 \boldsymbol{u}$ is concerned. However, this effect is too weak to be visible in Fig. 5, implying that the performance difference between LS and Taubin is mostly due to second order error $\Delta_2 \boldsymbol{u}$, in particular the last term of Eq. (43). □

4.3. Optimally Weighted Least Squares

A well known correction to LS is to appropriately weight each summand in Eq. (29) in the form

$$J = \sum_{\alpha=1}^{N} W_\alpha (\boldsymbol{\xi}_\alpha, \boldsymbol{u})^2, \tag{56}$$

which is minimized by the unit eigenvector of

$$\boldsymbol{M} = \sum_{\alpha=1}^{N} W_\alpha \boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top, \tag{57}$$

for the smallest eigenvalue. The weight W_α is determined so that the covariance matrix of the resulting estimate is as close to the KCR lower bound as possible.

Following the analysis in Section 4.1, we can easily see that the first order covariance matrix in Eq. (38) is now replaced by

$$V[\hat{\boldsymbol{u}}] = \varepsilon^2 \bar{\boldsymbol{M}}^- \Big(\sum_{\alpha=1}^{N} W_\alpha (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}) \bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\beta^\top \Big) \bar{\boldsymbol{M}}^-. \tag{58}$$

It is not difficult to see that this coincides with the KCR lower bound if we set

$$W_\alpha = \frac{1}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}. \tag{59}$$

In fact, we have

$$\begin{aligned} V[\hat{\boldsymbol{u}}] &= \varepsilon^2 \bar{\boldsymbol{M}}^- \Big(\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\beta^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} \Big) \bar{\boldsymbol{M}}^- \\ &= \varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{M}} \bar{\boldsymbol{M}}^- = \varepsilon^2 \bar{\boldsymbol{M}}^-, \end{aligned} \tag{60}$$

where we define

$$\bar{\boldsymbol{M}} = \sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}. \tag{61}$$

Evidently, Eq. (60) equals the KCR lower bound given by Eq. (17)).

However, we cannot use Eq. (59), because the true value $\boldsymbol{u}$ is unknown. So, we do iterations. Namely, we first give an appropriate initial guess of $\boldsymbol{u}$, say by LS, substitute it

into Eq. (59) and compute the eigenvector of the matrix $\boldsymbol{M}$ in Eq. (57) for the smallest eigenvalue. Using the resulting solution, we update the weight W_α, and this process is iterated. This method is known as *optimally weighted* (*iterative*) *least squares* or *reweight* procedure [26]. The fact that this method achieves the KCR lower bound to a first approximation was pointed out by Chernov and Lesort [5].

We now evaluate its accuracy. After the iterations have converged, the resulting solution $\hat{\boldsymbol{u}}$ satisfies

$$\hat{\boldsymbol{M}}\hat{\boldsymbol{u}} = \lambda\hat{\boldsymbol{u}}, \tag{62}$$

where

$$\hat{\boldsymbol{M}} = \sum_{\alpha=1}^{N} \frac{\boldsymbol{\xi}_\alpha \boldsymbol{\xi}_\alpha^\top}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})}. \tag{63}$$

Substituting $\boldsymbol{\xi}_\alpha = \bar{\boldsymbol{\xi}}_\alpha + \Delta\boldsymbol{\xi}_\alpha$, $\hat{\boldsymbol{u}} = \boldsymbol{u} + \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{u} + \cdots$, and $\lambda = \Delta_1\lambda + \Delta_2\lambda + \cdots$ into Eq. (62), we have

$$\begin{aligned}&(\bar{\boldsymbol{M}} + \Delta_1\boldsymbol{M} + \Delta_1^*\boldsymbol{M} + \Delta_2\boldsymbol{M} + \Delta_2^*\boldsymbol{M})\\ &\quad(\boldsymbol{u} + \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{u} + \cdots)\\ &\quad= (\Delta_1\lambda + \Delta_2\lambda + \cdots)(\boldsymbol{u} + \Delta_1\boldsymbol{u} + \Delta_2\boldsymbol{u} + \cdots),\end{aligned} \tag{64}$$

where we put

$$\Delta_1\boldsymbol{M} = \sum_{\alpha=1}^{N} \frac{\Delta\boldsymbol{\xi}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top + \bar{\boldsymbol{\xi}}_\alpha\Delta\boldsymbol{\xi}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}, \tag{65}$$

$$\Delta_2\boldsymbol{M} = \sum_{\alpha=1}^{N} \frac{\Delta\boldsymbol{\xi}_\alpha\Delta\boldsymbol{\xi}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}, \tag{66}$$

$$\Delta_1^*\boldsymbol{M} = -2\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}(\Delta_1\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}), \tag{67}$$

$$\begin{aligned}\Delta_2^*\boldsymbol{M} = &-2\sum_{\alpha=1}^{N} \frac{\Delta\boldsymbol{\xi}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top + \bar{\boldsymbol{\xi}}_\alpha\Delta\boldsymbol{\xi}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}(\Delta_1\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\\ &+\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\Big(-\frac{2(\Delta_2\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\\ &+\frac{4(\Delta_1\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2} - \frac{(\Delta_1\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\Delta_1\boldsymbol{u})}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\Big).\end{aligned} \tag{68}$$

Here, $\Delta_1^*\boldsymbol{M}$ and $\Delta_2^*\boldsymbol{M}$ are, respectively, the first and second order perturbations of $\boldsymbol{M}$ for using $\hat{\boldsymbol{u}}$ in the denominator in Eq. (63).

Equating first and second order terms on both sides of Eq. (64), we obtain

$$\bar{\boldsymbol{M}}\Delta_1\boldsymbol{u} + (\Delta_1\boldsymbol{M} + \Delta_1^*\boldsymbol{M})\boldsymbol{u} = \Delta_1\lambda\boldsymbol{u}, \tag{69}$$

$$\begin{aligned}&\bar{\boldsymbol{M}}\Delta_2\boldsymbol{u} + (\Delta_1\boldsymbol{M} + \Delta_1^*\boldsymbol{M})\Delta_1\boldsymbol{u} + (\Delta_2\boldsymbol{M} + \Delta_2^*\boldsymbol{M})\boldsymbol{u}\\ &= \Delta_1\lambda\Delta_1\boldsymbol{u} + \Delta_2\lambda\boldsymbol{u}.\end{aligned} \tag{70}$$

Computing the inner product with $\boldsymbol{u}$ on both sides of Eq. (69) and noting that $(\boldsymbol{u}, \bar{\boldsymbol{M}}\boldsymbol{u})$, $(\boldsymbol{u}, \Delta_1\boldsymbol{M}\boldsymbol{u})$, and $(\boldsymbol{u}, \Delta_1^*\boldsymbol{M}\boldsymbol{u})$ all identically vanish, we find that $\Delta_1\lambda = 0$. Multiplying $\bar{\boldsymbol{M}}^-$ on both sides of Eq. (69) and solving for $\Delta_1\boldsymbol{u}$, we obtain as before

$$\Delta_1\boldsymbol{u} = -\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\boldsymbol{u}, \tag{71}$$

whose covariance matrix coincides with the KCR lower bound, as shown earlier.

Multiplying $\bar{\boldsymbol{M}}^-$ on both sides of Eq. (70) and solving for $\Delta_2\boldsymbol{u}^\perp$ ($\equiv \bar{\boldsymbol{M}}^-\bar{\boldsymbol{M}}\Delta_2\boldsymbol{u}$), we obtain

$$\begin{aligned}&\Delta_2\boldsymbol{u}^\perp\\ &= -\bar{\boldsymbol{M}}^-(\Delta_1\boldsymbol{M} + \Delta_1^*\boldsymbol{M})\Delta_1\boldsymbol{u} - \bar{\boldsymbol{M}}^-(\Delta_2\boldsymbol{M} + \Delta_2^*\boldsymbol{M})\boldsymbol{u}\\ &= \bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\boldsymbol{u} + \bar{\boldsymbol{M}}^-\Delta_1^*\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\boldsymbol{u}\\ &\quad -\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{M}\boldsymbol{u} - \bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{M}\boldsymbol{u}.\end{aligned} \tag{72}$$

Now, we compute its expectation. We first see that

$$\begin{aligned}&E[\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\boldsymbol{u}]\\ &= E[\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N} \frac{\Delta\boldsymbol{\xi}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top + \bar{\boldsymbol{\xi}}_\alpha\Delta\boldsymbol{\xi}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}]\\ &= \bar{\boldsymbol{M}}^-\sum_{\alpha,\beta=1}^{N} \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta)E[\Delta\boldsymbol{\xi}_\alpha\Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})}\\ &\quad +\bar{\boldsymbol{M}}^-\sum_{\alpha,\beta=1}^{N} \frac{(\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\beta, E[\Delta\boldsymbol{\xi}_\alpha\Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})}\\ &= \varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}\\ &\quad +\varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}.\end{aligned} \tag{73}$$

We also see that

$$\begin{aligned}&E[\bar{\boldsymbol{M}}^-\Delta_1^*\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\boldsymbol{u}]\\ &= E[2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N} \frac{(\Delta_1\boldsymbol{M}\boldsymbol{u}, \bar{\boldsymbol{M}}^-V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}\\ &\quad \bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\boldsymbol{u}]\\ &= 2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}(V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u},\\ &\quad \bar{\boldsymbol{M}}^-E[(\Delta_1\boldsymbol{M}\boldsymbol{u})(\Delta_1\boldsymbol{M}\boldsymbol{u})^\top]\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha).\end{aligned} \tag{74}$$

The expectation $E[(\Delta_1\boldsymbol{M}\boldsymbol{u})(\Delta_1\boldsymbol{M}\boldsymbol{u})^\top]$ is

$$\begin{aligned}&E[(\Delta_1\boldsymbol{M}\boldsymbol{u})(\Delta_1\boldsymbol{M}\boldsymbol{u})^\top]\\ &= E[\sum_{\alpha=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}\sum_{\beta=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\beta, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\beta^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})}]\end{aligned}$$

$$= \sum_{\alpha,\beta=1}^{N} \frac{(\boldsymbol{u}, E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\beta^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})}$$
$$= \varepsilon^2 \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2} = \varepsilon^2 \sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}$$
$$= \varepsilon^2 \bar{\boldsymbol{M}}. \tag{75}$$

Hence, Eq. (74) becomes

$$E[\bar{\boldsymbol{M}}^- \Delta_1^* \boldsymbol{M} \bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M}\boldsymbol{u}]$$
$$= 2\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{M}} \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha)\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}$$
$$= 2\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha)\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}. \tag{76}$$

The expectation of $\bar{\boldsymbol{M}}^- \Delta_2 \boldsymbol{M}\boldsymbol{u}$ is

$$E[\bar{\boldsymbol{M}}^- \Delta_2 \boldsymbol{M}\boldsymbol{u}] = E[\bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{u})\Delta\boldsymbol{\xi}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}]$$
$$= \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} = \varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}$$
$$= \varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}}\boldsymbol{u}, \tag{77}$$

where we define

$$\bar{\boldsymbol{N}} = \sum_{\alpha=1}^{N} \frac{V_0[\boldsymbol{\xi}_\alpha]}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}. \tag{78}$$

The expectation of $\bar{\boldsymbol{M}}^- \Delta_2^* \boldsymbol{M}\boldsymbol{u}$ is

$$E[\bar{\boldsymbol{M}}^- \Delta_2^* \boldsymbol{M}\boldsymbol{u}]$$
$$= E[-2\bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\Delta_1\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}]$$
$$= 2\bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, E[\Delta\boldsymbol{\xi}_\alpha(\Delta_1\boldsymbol{M}\boldsymbol{u})^\top]\bar{\boldsymbol{M}}^- V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}. \tag{79}$$

The expectation $E[\Delta\boldsymbol{\xi}_\alpha(\Delta_1\boldsymbol{M}\boldsymbol{u})^\top]$ is

$$E[\Delta\boldsymbol{\xi}_\alpha(\Delta_1\boldsymbol{M}\boldsymbol{u})^\top] = E[\Delta\boldsymbol{\xi}_\alpha \sum_{\beta=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\beta, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\beta^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})}]$$
$$= \sum_{\beta=1}^{N} \frac{E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u}\bar{\boldsymbol{\xi}}_\beta^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})} = \frac{\varepsilon^2 V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}\bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}. \tag{80}$$

Hence, Eq. (79) becomes

$$E[\bar{\boldsymbol{M}}^- \Delta_2^* \boldsymbol{M}\boldsymbol{u}]$$

Table 2. The role of the Taubin method and renormalization.

	no weight		iterative reweight
eigenvalue problem	LS	$\leftrightarrow$	optimally weighted LS
	$\Downarrow$		$\Downarrow$
generalized eigenvalue problem	Taubin	$\leftrightarrow$	renormalization

$$= 2\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^3}$$
$$= 2\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}, \tag{81}$$

which is the same as Eq. (76). Thus, the expectation of $\Delta_2\boldsymbol{u}^\perp$ in Eq. (72)

$$E[\Delta_2\boldsymbol{u}^\perp]$$
$$= \varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}$$
$$+\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2} - \varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}}\boldsymbol{u}. \tag{82}$$

4.4. Renormalization

We can see the similarity between Eqs. (37) and (43) for (unweighted) LS and Eqs. (71) and (82) for optimally weighted LS, where the (unweighted) matrix $\boldsymbol{M}_0$ is replaced by the weighted matrix $\boldsymbol{M}$. We have also seen that the last term $-\varepsilon^2 \bar{\boldsymbol{M}}_0^- \boldsymbol{N}_0\boldsymbol{u}$ in Eq. (43) can be removed by using the Taubin method, replacing Eq. (31) by Eq. (45) by inserting the (unweighted) matrix $\boldsymbol{N}_0$. This implies that the last term $-\varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}}\boldsymbol{u}$ in Eq. (82) may be removed by replacing Eq. (62) by

$$\hat{\boldsymbol{M}}\hat{\boldsymbol{u}} = \lambda\hat{\boldsymbol{N}}, \tag{83}$$

by inserting the weighed matrix

$$\hat{\boldsymbol{N}} = \sum_{\alpha=1}^{N} \frac{V_0[\boldsymbol{\xi}_\alpha]}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})}. \tag{84}$$

Indeed, this is the idea of the *renormalization* of Kanatani [11, 12] (Table 2). His original idea was that the exact value $\boldsymbol{u}$ is obtained as the eigenvector of $\bar{\boldsymbol{M}}$ in Eq. (61) for eigenvalue 0. If we approximate $\bar{\boldsymbol{M}}$ by $\hat{\boldsymbol{M}}$ in Eq. (62), we have

$$\hat{\boldsymbol{M}} = \bar{\boldsymbol{M}} + \Delta_1\boldsymbol{M} + \Delta_1^*\boldsymbol{M} + \Delta_2\boldsymbol{M} + \Delta_2^*\boldsymbol{M}. \tag{85}$$

Evidently $E[\Delta_1 \boldsymbol{M}] = \boldsymbol{O}$ and $E[\Delta_1^* \boldsymbol{M}] = \boldsymbol{O}$, but we see from Eq. (66) that

$$E[\Delta_2 \boldsymbol{M}] = \sum_{\alpha=1}^{N} \frac{E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top]}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} = \sum_{\alpha=1}^{N} \frac{\varepsilon^2 V_0[\boldsymbol{\xi}_\alpha]}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} = \varepsilon^2 \bar{\boldsymbol{N}}. \tag{86}$$

Hence, $\hat{\boldsymbol{M}} - \varepsilon^2 \bar{\boldsymbol{N}}$ is closer to $\bar{\boldsymbol{M}}$ in expectation than $\hat{\boldsymbol{M}}$. Though we do not know ε^2 and $\bar{\boldsymbol{N}}$, the latter may be approximated by $\hat{\boldsymbol{N}}$. The former is simply regarded as an unknown to be estimated. Kanatani [11, 12] estimated it as the value c that make $\hat{\boldsymbol{M}} - c\hat{\boldsymbol{N}}$ singular, since the true value $\bar{\boldsymbol{M}}$ has eigenvalue 0. Thus, Kanatani's renormalization goes as follows:

1. Initialize $\hat{\boldsymbol{u}}$, say by LS, and let $c = 0$.
2. Solve the eigenvalue problem
$$(\hat{\boldsymbol{M}} - c\hat{\boldsymbol{N}})\boldsymbol{u} = \lambda \boldsymbol{u}, \tag{87}$$
and let $\boldsymbol{u}$ be the unit eigenvector for the eigenvalue λ closest to 0.
3. If $\lambda \approx 0$, return $\hat{\boldsymbol{u}}$ and stop. Else, let
$$c \leftarrow c + \frac{\lambda}{(\boldsymbol{u}, \hat{\boldsymbol{N}}\boldsymbol{u})}, \qquad \hat{\boldsymbol{u}} \leftarrow \boldsymbol{u}, \tag{88}$$
and go back to Step 2.

This method has been demonstrated to result in dramatic improvement over (unweighted or optimally weighted) LS in many computer vision problems including fundamental matrix computation for 3-D reconstruction and homography estimation for image mosaicing [17, 18]. We now analyze its accuracy.

After the iterations have converged, we have

$$(\hat{\boldsymbol{M}} - c\hat{\boldsymbol{N}})\hat{\boldsymbol{u}} = \boldsymbol{0}, \tag{89}$$

which is essentially Eq. (83). As before, we have the perturbation expansion

$$\Big(\bar{\boldsymbol{M}} + (\Delta_1 \boldsymbol{M} + \Delta_1^* \boldsymbol{M}) + (\Delta_2 \boldsymbol{M} + \Delta_2^* \boldsymbol{M}) + \cdots - (\Delta_1 c + \Delta_2 c + \cdots)(\bar{\boldsymbol{N}} + \Delta_1^* \boldsymbol{N} + \cdots)\Big)(\boldsymbol{u} + \Delta_1 \boldsymbol{u} + \Delta_2 \boldsymbol{u} + \cdots) = \boldsymbol{0}, \tag{90}$$

where

$$\Delta_1^* \boldsymbol{N} = -2 \sum_{\alpha=1}^{N} \frac{(\Delta_1 \boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}) V_0[\boldsymbol{\xi}_\alpha]}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}, \tag{91}$$

which arises from the expansion of the denominator in the expression of $\hat{\boldsymbol{N}}$ (the second order perturbation $\Delta_2^* \boldsymbol{N}$ does not affect the subsequent analysis).

Equating first and second order terms on both sides of Eq. (90), we obtain

$$\bar{\boldsymbol{M}} \Delta_1 \boldsymbol{u} + (\Delta_1 \boldsymbol{M} + \Delta_1^* \boldsymbol{M} - \Delta_1 c \bar{\boldsymbol{N}})\boldsymbol{u} = \boldsymbol{0}, \tag{92}$$

$$\bar{\boldsymbol{M}} \Delta_2 \boldsymbol{u} + (\Delta_1 \boldsymbol{M} + \Delta_1^* \boldsymbol{M} - \Delta_1 c \bar{\boldsymbol{N}})\Delta_1 \boldsymbol{u} + (\Delta_2 \boldsymbol{M} + \Delta_2^* \boldsymbol{M} - \Delta_1 c \Delta_1^* \boldsymbol{N} - \Delta_2 c \bar{\boldsymbol{N}})\boldsymbol{u} = \boldsymbol{0}. \tag{93}$$

Computing the inner product with $\hat{\boldsymbol{u}}$ on both sides of Eq. (92), we find that $\Delta_1 c = 0$ as before. Multiplying $\bar{\boldsymbol{M}}^-$ on both sides of Eq. (92) and solving for $\Delta_1 \boldsymbol{u}$, we again obtain Eq. (71).

Multiplying $\bar{\boldsymbol{M}}^-$ on both sides of Eq. (93) and solving for $\Delta_2 \boldsymbol{u}^\perp$, we obtain

$$\begin{aligned}
\Delta_2 \boldsymbol{u}^\perp &= -\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \Delta_1 \boldsymbol{u} - \bar{\boldsymbol{M}}^- \Delta_1^* \boldsymbol{M} \Delta_1 \boldsymbol{u} - \bar{\boldsymbol{M}}^- \Delta_2 \boldsymbol{M} \boldsymbol{u} \\
&\quad - \bar{\boldsymbol{M}}^- \Delta_2^* \boldsymbol{M} \boldsymbol{u} + \Delta_2 c \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}} \boldsymbol{u} \\
&= \bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u} + \bar{\boldsymbol{M}}^- \Delta_1^* \boldsymbol{M} \bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u} \\
&\quad - \bar{\boldsymbol{M}}^- \Delta_2 \boldsymbol{M} \boldsymbol{u} - \bar{\boldsymbol{M}}^- \Delta_2^* \boldsymbol{M} \boldsymbol{u} + \Delta_2 c \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}} \boldsymbol{u}.
\end{aligned} \tag{94}$$

Comparing this with Eq. (72), we find that an extra term, $\Delta_2 c \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}} \boldsymbol{u}$, is added. We now evaluate its expectation.

Computing the inner product with $\boldsymbol{u}$ on both sides of Eq. (93) and noting that $(\boldsymbol{u}, \bar{\boldsymbol{M}} \Delta_2 \boldsymbol{u})$, $(\boldsymbol{u}, \Delta_1^* \boldsymbol{M} \Delta_1 \boldsymbol{u})$, and $(\boldsymbol{u}, \Delta_2^* \boldsymbol{M} \boldsymbol{u})$ all identically vanish, we have

$$\Delta_2 c = \frac{(\boldsymbol{u}, \Delta_2 \boldsymbol{M} \boldsymbol{u}) - (\boldsymbol{u}, \Delta_1 \boldsymbol{M} \Delta_1 \boldsymbol{u})}{(\boldsymbol{u}, \bar{\boldsymbol{N}} \boldsymbol{u})} \tag{95}$$

We first note from the definition of $\bar{\boldsymbol{N}}$ in Eq. (78) that

$$(\boldsymbol{u}, \bar{\boldsymbol{N}} \boldsymbol{u}) = \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} = N. \tag{96}$$

The expectation of $(\boldsymbol{u}, \Delta_2 \boldsymbol{M} \boldsymbol{u})$ is

$$\begin{aligned}
&E[(\boldsymbol{u}, \Delta_2 \boldsymbol{M} \boldsymbol{u})] \\
&= \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\alpha^\top]\boldsymbol{u})}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} = \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, \varepsilon^2 V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} \\
&= N\varepsilon^2.
\end{aligned} \tag{97}$$

The expectation of $(\boldsymbol{u}, \Delta_1 \boldsymbol{M} \Delta_1 \boldsymbol{u})$ is

$$\begin{aligned}
&E[(\boldsymbol{u}, \Delta_1 \boldsymbol{M} \Delta_1 \boldsymbol{u})] \\
&= E[(\boldsymbol{u}, \Delta_1 \boldsymbol{M} \bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u})] \\
&= E[(\Delta_1 \boldsymbol{M} \boldsymbol{u}, \bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u})] \\
&= E[(\sum_{\alpha=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\alpha, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}, \bar{\boldsymbol{M}}^- \sum_{\beta=1}^{N} \frac{(\Delta\boldsymbol{\xi}_\beta, \boldsymbol{u})\bar{\boldsymbol{\xi}}_\beta}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})})]
\end{aligned}$$

$$= \sum_{\alpha,\beta=1}^{N} \frac{(\boldsymbol{u}, E[\Delta\boldsymbol{\xi}_\alpha \Delta\boldsymbol{\xi}_\beta^\top]\boldsymbol{u})(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\beta)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\beta]\boldsymbol{u})}$$

$$= \varepsilon^2 \sum_{\alpha=1}^{N} \frac{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}$$

$$= \varepsilon^2 \sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha)}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} = \varepsilon^2 \sum_{\alpha=1}^{N} \frac{\mathrm{tr}[\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top]}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}$$

$$= \varepsilon^2 \mathrm{tr}[\bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{\bar{\boldsymbol{\xi}}_\alpha \bar{\boldsymbol{\xi}}_\alpha^\top}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}] = \varepsilon^2 \mathrm{tr}[\bar{\boldsymbol{M}}^- \bar{\boldsymbol{M}}]$$

$$= \varepsilon^2 \mathrm{tr}[\boldsymbol{P}_{\mathbf{u}}] = (p-1)\varepsilon^2, \tag{98}$$

where p is the dimension of the parameter vector $\boldsymbol{u}$. Thus, from Eq. (95) we have

$$E[\Delta_2 c] = \Big(1 - \frac{p-1}{N}\Big)\varepsilon^2, \tag{99}$$

and hence from Eq. (94)

$$E[\Delta_2 \boldsymbol{u}^\perp] = \varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}$$
$$+\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}$$
$$-\frac{p-1}{N}\varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}}\boldsymbol{u}. \tag{100}$$

Eq. (99) corresponds to the well known formula of unbiased estimation of the noise variance ε^2 (note that the p-dimensional unit vector $\boldsymbol{u}$ has $p-1$ degrees of freedom).

If the number N of data is fairly large, which is the case in many vision applications, the last term in Eq. (100) is insignificant, resulting in the frequently reported dramatic improvement over optimally weighted LS.

Kanatani's renormalization was at first not well understood. This was due to the generally held preconception that parameter estimation should be done by minimizing something. People wondered what renormalization was actually minimizing. In this line of thought, Chojnacki et al. [6] interpreted it to be an approximation to maximum likelihood estimation. We have seen, however, that optimal estimation does not necessarily mean minimization and that renormalization is an effort to improve accuracy by a direct means.

Example 4 Figure 6 is the RMS error plot corresponding to Fig. 5 using the ellipse data in Example 3. The thick solid line is for LS, the dashed line is for optimally weighted LS, and the thick solid line is for renormalization. The dotted line is for the KCR lower bound. Although the plots for optimally weighted LS and renormalization should both be tangent to that of the KCR lower bound at $\sigma = 0$, but not for LS, this is not visible from the figure, again confirming

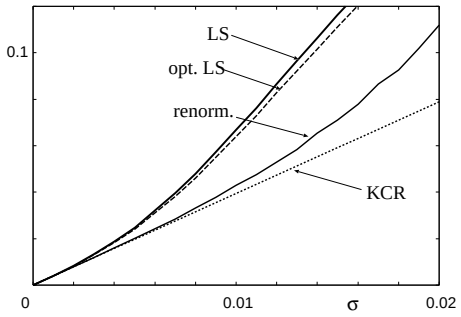

Figure 6. Noise level vs. RMS error for the ellipse data in Fig. 4: LS (thick solid line), optimally weighted LS (dashed line), renormalization (thin solid line), and the KCR lower bound (dotted line).

that the performance difference is mostly due to the second order error $\Delta_2\boldsymbol{u}$.

In fact, we can see from Fig. 6 that the accuracy gain of optimally weighted LS over the (unweighted) LS is rather small, meaning that satisfaction of the KCR lower bound in the first order is not a good indicator of high accuracy.

In contrast, renormalization performs considerably better than optimally weighted LS, clearly demonstrating that the last term of Eq. (82) has a decisive influence on the accuracy. The situation is similar to the relationship between LS and the Taubin method. □

4.5. Maximum Likelihood Estimation

Maximum likelihood estimation (*ML*) minimizes Eq. (14), which reduces for the linearized constraint of Eq. (16) to

$$J = \sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \boldsymbol{u})^2}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})}. \tag{101}$$

Differentiating this with respect to $\boldsymbol{u}$, we obtain

$$\nabla_{\mathbf{u}} J = \sum_{\alpha=1}^{N} \frac{2(\boldsymbol{\xi}_\alpha, \boldsymbol{u})\boldsymbol{\xi}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})} - \sum_{\alpha=1}^{N} \frac{2(\boldsymbol{\xi}_\alpha, \boldsymbol{u})^2 V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}. \tag{102}$$

Hence, the ML estimator $\hat{\boldsymbol{u}}$ is the solution of

$$\hat{\boldsymbol{M}}\hat{\boldsymbol{u}} = \hat{\boldsymbol{L}}\hat{\boldsymbol{u}}, \tag{103}$$

where $\hat{\boldsymbol{M}}$ is defined by Eq. (63) and $\hat{\boldsymbol{L}}$ is given by

$$\hat{\boldsymbol{L}} = \sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \hat{\boldsymbol{u}})^2 V_0[\boldsymbol{\xi}_\alpha]}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})^2}. \tag{104}$$

Equation (103) can be solved using various numerical schemes. The *FNS* (*fundamental numerical scheme*) of Chojnacki et al. [7] reduces Eq. (103) to iterative eigenvalue

problem solving (see Appendix E), and the *HEIV* (*heteroscedastic errors-in-variable*) of Leedan and Meer [20] reduces it to iterative generalized eigenvalue problem solving (see Appendix F). We may also do Gauss-Newton iterations directly (see Appendix G). We now analyze the accuracy of the ML estimator.

Whatever iterative scheme is used, Eq. (103) holds after the iterations have converged. The perturbation expansion of Eq. (103) is

$$\begin{aligned}&(\bar{\boldsymbol{M}}+\Delta_1\boldsymbol{M}+\Delta_1^*\boldsymbol{M}+\Delta_2\boldsymbol{M}+\Delta_2^*\boldsymbol{M}+\cdots\\&-\Delta_2\boldsymbol{L}-\Delta_2^*\boldsymbol{L})(\bar{\boldsymbol{u}}+\Delta_1\boldsymbol{u}+\Delta_2\boldsymbol{u}+\cdots)=\boldsymbol{0},\end{aligned}\tag{105}$$

where

$$\begin{aligned}\Delta_2\boldsymbol{L}&=\sum_{\alpha=1}^{N}\frac{(\Delta\bar{\boldsymbol{\xi}}_\alpha,\bar{\boldsymbol{u}})^2V_0[\boldsymbol{\xi}_\alpha]}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2},\\\Delta_2^*\boldsymbol{L}&=\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,\Delta_1\boldsymbol{u})^2V_0[\boldsymbol{\xi}_\alpha]}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}\\&\quad+2\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,\Delta_1\boldsymbol{u})(\Delta\bar{\boldsymbol{\xi}}_\alpha,\bar{\boldsymbol{u}})V_0[\boldsymbol{\xi}_\alpha]}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}.\end{aligned}\tag{106}$$

Note that Eq. (104) vanishes if $\boldsymbol{\xi}_\alpha$ and $\hat{\boldsymbol{u}}$ are replaced by $\bar{\boldsymbol{\xi}}_\alpha$ and $\boldsymbol{u}$, respectively. Hence, the 0th order term of $\boldsymbol{L}$ is $\boldsymbol{O}$. Since Eq. (104) contains the quadratic term $(\boldsymbol{\xi}_\alpha,\hat{\boldsymbol{u}})^2$, the first order perturbations $\Delta_1\boldsymbol{L}$ and $\Delta_1^*\boldsymbol{L}$ are also $\boldsymbol{O}$.

Equating first and second order terms on both sides of Eq. (106), we obtain

$$\bar{\boldsymbol{M}}\Delta_1\boldsymbol{u}+(\Delta_1\boldsymbol{M}+\Delta_1^*\boldsymbol{M})\bar{\boldsymbol{u}}=\boldsymbol{0},\tag{107}$$

$$\begin{aligned}&\bar{\boldsymbol{M}}\Delta_2\boldsymbol{u}+(\Delta_1\boldsymbol{M}+\Delta_1^*\boldsymbol{M})\Delta_1\boldsymbol{u}+(\Delta_2\boldsymbol{M}+\Delta_2^*\boldsymbol{M}\\&-\Delta_2\boldsymbol{L}-\Delta_2^*\boldsymbol{L})\bar{\boldsymbol{u}}=\boldsymbol{0}.\end{aligned}\tag{108}$$

Multiplying $\bar{\boldsymbol{M}}^-$ on both sides of Eq. (107) and solving for $\Delta_1\boldsymbol{u}$, we again obtain Eq. (71).

Multiplying $\bar{\boldsymbol{M}}^-$ on both sides of Eq. (108) and solving for $\Delta_2\boldsymbol{u}^\perp$, we obtain

$$\begin{aligned}&\Delta_2\boldsymbol{u}^\perp\\&=-\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\Delta_1\boldsymbol{u}-\bar{\boldsymbol{M}}^-\Delta_1^*\boldsymbol{M}\Delta_1\boldsymbol{u}-\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{M}\bar{\boldsymbol{u}}\\&\quad-\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{M}\bar{\boldsymbol{u}}+\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{L}\bar{\boldsymbol{u}}+\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{L}\bar{\boldsymbol{u}}\\&=\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{u}}+\bar{\boldsymbol{M}}^-\Delta_1^*\boldsymbol{M}\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{u}}\\&\quad-\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{M}\bar{\boldsymbol{u}}-\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{M}\bar{\boldsymbol{u}}+\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{L}\bar{\boldsymbol{u}}\\&\quad+\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{L}\bar{\boldsymbol{u}}\end{aligned}\tag{109}$$

For computing its expectation, we only need to consider the new terms $\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{L}\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{L}\bar{\boldsymbol{u}}$. First, we see that

$$\begin{aligned}&E[\bar{\boldsymbol{M}}^-\Delta_2\boldsymbol{L}\bar{\boldsymbol{u}}]\\&=\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{u}},E[\Delta\bar{\boldsymbol{\xi}}_\alpha\Delta\bar{\boldsymbol{\xi}}_\alpha^\top]\bar{\boldsymbol{u}})V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}\\&=\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{u}},\varepsilon^2V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}\\&=\varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})}=\varepsilon^2\bar{\boldsymbol{M}}^-\bar{\boldsymbol{N}}\bar{\boldsymbol{u}}.\end{aligned}\tag{110}$$

For $\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{L}\bar{\boldsymbol{u}}$, we have

$$\begin{aligned}&E[\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{L}\bar{\boldsymbol{u}}]\\&=\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,E[\Delta_1\boldsymbol{u}\Delta_1\boldsymbol{u}^\top]\bar{\boldsymbol{\xi}}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}\\&\quad+2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,E[\Delta_1\boldsymbol{u}\Delta\bar{\boldsymbol{\xi}}_\alpha^\top]\bar{\boldsymbol{u}})V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}.\end{aligned}\tag{111}$$

We have already seen that the first order error $\Delta_1\boldsymbol{u}$ satisfies the KCR lower bound, so $E[\Delta_1\boldsymbol{u}\Delta_1\boldsymbol{u}^\top]=\varepsilon\bar{\boldsymbol{M}}^-$ (see Eq. (60)). On the other hand,

$$\begin{aligned}&E[\Delta_1\boldsymbol{u}\Delta\bar{\boldsymbol{\xi}}_\alpha^\top]\bar{\boldsymbol{u}}\\&=-E[\bar{\boldsymbol{M}}^-\Delta_1\boldsymbol{M}\bar{\boldsymbol{u}}\Delta\bar{\boldsymbol{\xi}}_\alpha^\top]\bar{\boldsymbol{u}}\\&=-\bar{\boldsymbol{M}}^-E[\sum_{\beta=1}^{N}\frac{\Delta\boldsymbol{\xi}_\alpha\bar{\boldsymbol{\xi}}_\alpha^\top+\bar{\boldsymbol{\xi}}_\alpha\Delta\boldsymbol{\xi}_\alpha^\top}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\beta]\bar{\boldsymbol{u}})}\bar{\boldsymbol{u}}\Delta\bar{\boldsymbol{\xi}}_\alpha^\top]\\&=-\bar{\boldsymbol{M}}^-\sum_{\beta=1}^{N}\frac{(\bar{\boldsymbol{u}},E[\Delta\boldsymbol{\xi}_\beta\Delta\bar{\boldsymbol{\xi}}_\alpha^\top]\bar{\boldsymbol{u}})\bar{\boldsymbol{\xi}}_\beta}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\beta]\bar{\boldsymbol{u}})}\\&=-\varepsilon^2\bar{\boldsymbol{M}}^-\frac{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})\bar{\boldsymbol{\xi}}_\alpha}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})}=-\varepsilon^2\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha.\end{aligned}\tag{112}$$

Hence,

$$\begin{aligned}&E[\bar{\boldsymbol{M}}^-\Delta_2^*\boldsymbol{L}\bar{\boldsymbol{u}}]\\&=\varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}\\&\quad-2\varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,\bar{\boldsymbol{M}}^-\boldsymbol{\xi}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}.\\&=-\varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{\xi}}_\alpha,\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha)V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}}}{(\bar{\boldsymbol{u}},V_0[\boldsymbol{\xi}_\alpha]\bar{\boldsymbol{u}})^2}.\end{aligned}\tag{113}$$

Adding Eqs. (110) and (113) to Eq. (82), we conclude that

$$E[\Delta_2\boldsymbol{u}^\perp]=\varepsilon^2\bar{\boldsymbol{M}}^-\sum_{\alpha=1}^{N}\frac{(\bar{\boldsymbol{M}}^-\bar{\boldsymbol{\xi}}_\alpha,V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})\bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u},V_0[\boldsymbol{\xi}_\alpha]\boldsymbol{u})^2}.\tag{114}$$

Comparing this with Eqs. (82) and (100), we can see the last two terms there are removed.

There has been a widespread misunderstanding that optimally weighted LS can actually compute ML because Eq. (56) is identical to Eq. (101) if the weight W_α is chosen as in Eq. (59). However, this is not so [7, 12]. The important thing is not *what* to minimize but *how* it is minimized.

Optimally weighted LS minimizes Eq. (101) for $\boldsymbol{u}$ in the numerator with $\boldsymbol{u}$ in the denominator fixed. Then, the resulting solution $\boldsymbol{u}$ is substituted into the denominator, Eq. (101) is minimized for $\boldsymbol{u}$ in the numerator, and this is iterated. This means that when the solution $\hat{\boldsymbol{u}}$ is obtained, it is guaranteed that

$$\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \hat{\boldsymbol{u}} + \delta\boldsymbol{u})^2}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})} \geq \sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \hat{\boldsymbol{u}})^2}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})}, \tag{115}$$

for any infinitesimal perturbation $\delta\boldsymbol{u}$, which the convergence of optimally weighted LS means. This, however, does not guarantee that

$$\sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \hat{\boldsymbol{u}} + \delta\boldsymbol{u})^2}{(\hat{\boldsymbol{u}} + \delta\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha](\hat{\boldsymbol{u}} + \delta\boldsymbol{u}))} \geq \sum_{\alpha=1}^{N} \frac{(\boldsymbol{\xi}_\alpha, \hat{\boldsymbol{u}})^2}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})}, \tag{116}$$

for any infinitesimal perturbation $\delta\boldsymbol{u}$, which minimization of J really means. The difference between Eq. (115) and Eq. (116) is very large: the latter eliminates the last two terms of $E[\Delta_1\boldsymbol{u}^\perp]$ in Eq. (82). Renormalization is intermediate in the sense that it eliminates only the last term (almost).

4.6. Hyperaccuracy Fitting

It has been widely believed that ML is the best method of all. Indeed, no method has been found that outperforms ML, aside from the semiparametric approach in the asymptotic limit $N \to \infty$ (cf. Section 2.4).

However, Eq. (114) implies the possibility of improving the accuracy of ML further. Namely, we "subtract" Eq. (114) from the ML estimator $\hat{\boldsymbol{u}}$. Of course, Eq. (114) cannot be precisely computed, because it involves the true values $\bar{\boldsymbol{\xi}}_\alpha$ and $\boldsymbol{u}$. So, we approximate them by the data $\boldsymbol{\xi}_\alpha$ and the ML estimator $\hat{\boldsymbol{u}}$. As is well known in statistics (e.g., see [12]), the unknown squared noise level ε^2 is estimated from the residual of Eq. (101) in the following form:

$$\hat{\varepsilon}^2 = \frac{(\hat{\boldsymbol{u}}, \hat{\boldsymbol{M}}\hat{\boldsymbol{u}})}{N - (p-1)}. \tag{117}$$

Thus, the correction has the form

$$\tilde{\boldsymbol{u}} = N[\hat{\boldsymbol{u}} - \hat{\varepsilon}^2 \hat{\boldsymbol{M}}^- \sum_{\alpha=1}^{N} \frac{(\hat{\boldsymbol{M}}^- \boldsymbol{\xi}_\alpha, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})\boldsymbol{\xi}_\alpha}{(\hat{\boldsymbol{u}}, V_0[\boldsymbol{\xi}_\alpha]\hat{\boldsymbol{u}})^2}], \tag{118}$$

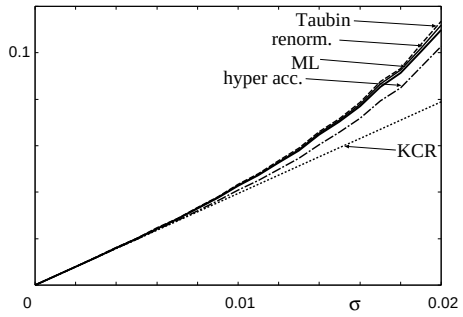

Figure 7. Noise level vs. RMS error for the ellipse data in Fig. 4: Taubin (dashed line), renormalization (thin solid line), ML (thick solid line), hyperaccurate correction (chained line), and the KCR lower bound (dotted line).

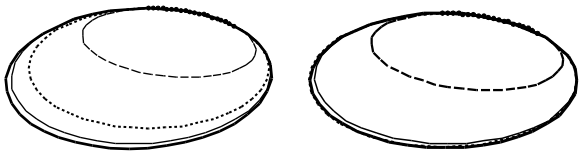

Figure 8. Two instances of ellipse fitting: LS (broken line), ML (thick solid line), hyperaccuracy correction (thin solid line), true ellipse (dotted line).

where the operation $N[\,\cdot\,]$ in denotes normalization to unit norm for compensating for the parallel component $\Delta\boldsymbol{u}^\parallel$ (see Fig. 3).

Example 5 Figure 7 shows the RMS error plot corresponding to Figs. 5 and 6 using the ellipse data in the Example 3. The dashed line is for the Taubin method, the thin line is for renormalization, and the thick solid line is for ML; we used the FNS of Chojnacki et al. [7] for computing ML. The dotted line is for the KCR lower bound. We can see that the Taubin method, renormalization, and ML have comparable accuracy. Their error plots almost agree with the KCR lower bound when the noise level is low. As the noise increases, however, their difference from the KCR lower bound slowly grows.

The chained line shows the result of the hyperaccurate correction of Eq. (118). We can see that the error is reduced by this correction and that the resulting RMS almost coincides with the KCR lower bound[2].

Figure 8(a) shows one instance of ellipse fitting (σ = 0.015). The dotted line shows the true ellipse; the broken line is for LS; the thick solid line is for ML; the thin solid

[2]The hyperaccuracy correction of ellipse fitting was first presented in [16], but the term $\Delta_2^* \boldsymbol{L}$ was not taken into account.

Table 3. Average error ratio of different methods.

LS	1.636
Optimally weighted LS	1.575
Taubin	1.144
Renormalization	1.133
ML	1.125
Hyperaccurate correction	1.007
KCR lower bound	1.000

line is for the hyperaccurate correction. We can see that the fitted ellipse is closer to the true shape after the correction. Figure 8(b) is another instance ($\sigma = 0.015$). In this case, the ellipse given by ML is already very accurate, and it slightly deviates from the true shape after the correction.

Thus, the accuracy sometimes improves and sometimes deteriorates. Overall, however, the cases of improvement is the majority; on average we observe slight improvement as shown in Fig. 7. Closely examining many examples, we have observed that the accuracy drop occurs almost always when the ML fitted ellipse falls inside the true shape. However, the majority of the fitted ellipses are outside the true shape. Thus, the correction is effective on average.

We infer that ML is likely to produce ellipses outside the true shape because it is parameterized in the form of Eq. (20). If the major or minor axis of the ellipse is a, the coefficient of x^2 or y^2 is proportional to $1/a^2$. If $1/a^2$ is "unbiased", a is biased to be larger than the true value, as can be easily seen from the shape of the graph of $y = 1/x^2$.

For comparing all the methods tested so far, we define the "error ratio" $D/D_{\rm KCR}$ of D in Eq. (54) divided by $D_{\rm KCR}$ in Eq. (55) and average it over the tested range of σ. Table 3 list this value for different method. □

5. Conclusions

We have given a rigorous accuracy analysis of various techniques for geometric fitting. We first pointed out how our problem is different from traditional statistical analysis and explained why we need a different framework. After giving general theories in our new framework, we selected typical existing techniques and analytically evaluated their accuracy up to second order terms. Table 4 summarizes the first order error and the second order bias. We also conducted numerical simulations of ellipse fitting. Our main observations are as follows:

1. LS and the Taubin method have the same error to a first approximation. However, the latter achieves much higher accuracy, because a dominant second order bias term of LS is removed.
2. Optimally weighted LS achieves the KCR lower bound to a first approximation. However, the accuracy gain over (unweighted) LS is rather small. This is due to the existence of second order bias terms.
3. Renormalization nearly removes the dominant bias term of optimally weighted LS, resulting in considerable accuracy improvement.
4. ML is less biased than renormalization. However, the accuracy gain is rather small.
5. By estimating and subtracting the bias term from the ML solution, we can achieve higher accuracy than ML ("hyperaccuracy").

Thus, we conclude that it is the *second order bias terms*, not the variance of the first order error, that have dominant effects over the accuracy. However, not all terms have the same degree of influence. Detailed evaluation of this requires further investigation.

Acknowledgments: This work was supported in part by the Ministry of Education, Culture, Sports, Science and Technology, Japan, under the Grant in Aid for Scientific Research C(2) (No. 15500113). The author thanks Wojciech Chojnacki of the University of Adelaide, Australia, and Nikolai Chernov of the University of Alabama at Birmingham, U.S.A. for helpful discussions. He also thanks Junpei Yamada of Okayama University for help doing numerical experiments.

References

[1] H. Akaike, A new look at the statistical model identification, IEEE Trans. Autom. Control, **16**-6 (1977), 716–723.

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

[3] E. Begelfor and M. Werman, How to put probabilities on homographies, *IEEE Trans. Patt. Anal. Mach. Intell.,* **27**-10 (2005-10), 1666-1670.

[4] 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.

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

[6] W. Chojnacki, M. J. Brooks and A. van den Hengel, Rationalising the renormalisation method of Kanatani, *J. Math. Imaging Vision*, **14**-1 (2001), 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), 1294–1303.

[8] W. Chojnacki, M. J. Brooks, A. van den Hengel and D. Gawley, From FNS to HEIV: A link between two vision parameter estimation methods, *IEEE Trans. Patt. Anal. Mach. Intell.*, **26**-2 (2004-2), 264–268.

[9] W. Chojnacki, M. J. Brooks, A. van den Hengel and D. Gawley, FNS, CFNS and HEIV: A unifying approach, *J. Math. Imaging Vision*, **23**-2 (2005-9), 175–183.

Table 4. Summary of the first order error and the second order bias.

method	first order error	second order bias
LS	$-\bar{\boldsymbol{M}}_0^- \Delta_1 \boldsymbol{M}_0 \boldsymbol{u}$	$\varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^N (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) \bar{\boldsymbol{\xi}}_\alpha + \varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^N (\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u} - \varepsilon^2 \bar{\boldsymbol{M}}_0^- \boldsymbol{N}_0 \boldsymbol{u}$
Taubin	$-\bar{\boldsymbol{M}}_0^- \Delta_1 \boldsymbol{M}_0 \boldsymbol{u}$	$\varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^N (\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) \bar{\boldsymbol{\xi}}_\alpha + \varepsilon^2 \bar{\boldsymbol{M}}_0^- \sum_{\alpha=1}^N (\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}_0^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}$
opt. LS	$-\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u}$	$\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^N \frac{(\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}) \bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u})^2} + \varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^N \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u})^2} - \varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}} \boldsymbol{u}$
renormalization	$-\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u}$	$\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^N \frac{(\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}) \bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u})^2} + \varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^N \frac{(\bar{\boldsymbol{\xi}}_\alpha, \bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha) V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u})^2} - \frac{p-1}{N} \varepsilon^2 \bar{\boldsymbol{M}}^- \bar{\boldsymbol{N}} \boldsymbol{u}$
ML	$-\bar{\boldsymbol{M}}^- \Delta_1 \boldsymbol{M} \boldsymbol{u}$	$\varepsilon^2 \bar{\boldsymbol{M}}^- \sum_{\alpha=1}^N \frac{(\bar{\boldsymbol{M}}^- \bar{\boldsymbol{\xi}}_\alpha, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u}) \bar{\boldsymbol{\xi}}_\alpha}{(\boldsymbol{u}, V_0[\boldsymbol{\xi}_\alpha] \boldsymbol{u})^2}$

[10] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, U.K., 2000.

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

[12] K. Kanatani, *Statistical Optimization for Geometric Computation: Theory and Practice*, Elsevier, Amsterdam, The Netherlands, 1996; Reprinted by Dover, New York, U.S.A., 2005.

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

[14] K. Kanatani, Uncertainty modeling and model selection for geometric inference, *IEEE Trans. Patt. Anal. Machine Intell.*, **26**-10 (2004), 1307–1319.

[15] K. Kanatani, Further improving geometric fitting, *Proc. 5th Int. Conf. 3-D Digital Imaging and Modeling (3DIM2005)*, June 2005, Ottawa, Canada, pp. 2–13.

[16] K. Kanatani, Ellipse fitting with hyperaccuracy, *Proc. 9th Euro Conf. Comput. Vision*, May 2006, Graz, Austria, Vol. 1, pp. 484–495.

[17] K. Kanatani and N. Ohta, Comparing optimal three-dimensional reconstruction for finite motion and optical flow, *J. Electronic Imaging*, **12**-3 (2003), 478–488.

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

[19] K. Kanatani and Y. Sugaya, High accuracy fundamental matrix computation and its performance evaluations, *Proc. 17th British Machine Vision Conf.*, September 2006, Edinburgh, U.K., to appear.

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

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

[22] N. Ohta, Motion parameter estimation from optical flow without nuisance parameters, *3rd Int. Workshop on Statistical and Computational Theory of Vision*, October 2003, Nice, France: `http://www.stat.ucla.edu/~sczhu/Workshops/SCTV2003.html`

[23] 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. Vision Pattern Recog.*, June 2003, Madison, WI, U.S.A., Vol. 1, pp. 432–439.

[24] X. Pennec, P. Fillard and N. Ayache, A Riemannian framework for tensor computing, *Int. J. Comput. Vision*, **66**-1 (2006-1), 41–66

[25] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*, World Scientific, Singapore, 1989.

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

Appendix

A: Derivation of the KCR Lower Bound

For simplicity, we consider only the case where no intrinsic constraints exist on the data $\boldsymbol{x}_\alpha$ or the parameter $\boldsymbol{u}$ and the noise is identical and isotropic Gaussian with mean 0 and variance ε^2. In other words, we assume that the probability distribution density of each datum $\boldsymbol{x}_\alpha$ is

$$p(\boldsymbol{x}_\alpha) = \frac{1}{(\sqrt{2\pi})^n \varepsilon^n} e^{-\|\mathbf{x}_\alpha - \bar{\mathbf{x}}_\alpha\|^2/2\varepsilon^2}. \tag{119}$$

Suppose an unbiased estimator $\hat{\boldsymbol{u}}(\boldsymbol{x}_1, ..., \boldsymbol{x}_N)$ is given. Its unbiasedness mean

$$E[\hat{\boldsymbol{u}} - \boldsymbol{u}] = \boldsymbol{0}, \tag{120}$$

where $E[\,\cdot\,]$ is expectation over the joint probability density $p(\boldsymbol{x}_1) \cdots p(\boldsymbol{x}_N)$. Since this density is parameterized by the

true data values $\bar{\boldsymbol{x}}_\alpha$, Eq. (120) can be viewed as an equation of $\bar{\boldsymbol{x}}_\alpha$ as well as the unknown $\boldsymbol{u}$. The crucial fact is that Eq. (120) should be an *identity* in $\bar{\boldsymbol{x}}_\alpha$ and $\boldsymbol{u}$ that satisfies Eq. (1), because unbiasedness is a "property" of the estimator $\hat{\boldsymbol{u}}$ that should hold for whatever values of $\bar{\boldsymbol{x}}_\alpha$ and $\boldsymbol{u}$. Hence, Eq. (120) should be invariant to infinitesimal variation of $\bar{\boldsymbol{x}}_\alpha$ and $\boldsymbol{u}$. This means

$$\begin{aligned}\delta \int (\hat{\boldsymbol{u}} - \boldsymbol{u}) p_1 \cdots p_N d\boldsymbol{x} &= -\int (\delta \boldsymbol{u}) p_1 \cdots p_N d\boldsymbol{x} \\ &+ \sum_{\alpha=1}^{N} \int (\hat{\boldsymbol{u}} - \boldsymbol{u}) p_1 \cdots \delta p_\alpha \cdots p_N d\boldsymbol{x} \\ = \; -\delta \boldsymbol{u} &+ \int (\hat{\boldsymbol{u}} - \boldsymbol{u}) \sum_{\alpha=1}^{N} (p_1 \cdots \delta p_\alpha \cdots p_N) d\boldsymbol{x}, \quad (121)\end{aligned}$$

where p_α is an abbreviation of $p(\boldsymbol{x}_\alpha)$ and $\int d\boldsymbol{x}$ is a shorthand of $\int \cdots \int d\boldsymbol{x}_1 \cdots \boldsymbol{x}_N$. Note that we consider variations in $\bar{\boldsymbol{x}}_\alpha$ (not $\boldsymbol{x}_\alpha$) and $\boldsymbol{u}$. Since the estimator $\hat{\boldsymbol{u}}$ is a function of the data $\boldsymbol{x}_\alpha$, it does not change for these variations. The variation $\delta \boldsymbol{u}$ is independent of $\boldsymbol{x}_\alpha$, so it can be moved outside the integral $\int d\boldsymbol{x}$. Also note that $\int p_1 \cdots p_N d\boldsymbol{x} = 1$.

The infinitesimal variation of Eq. (119) with respect to $\bar{\boldsymbol{x}}_\alpha$ is

$$\delta p_\alpha = (\boldsymbol{l}_\alpha, \delta \bar{\boldsymbol{x}}_\alpha) p_\alpha, \quad (122)$$

where we define the *score* $\boldsymbol{l}_\alpha$ by

$$\boldsymbol{l}_\alpha \equiv \nabla_{\bar{\mathbf{x}}_\alpha} \log p_\alpha = \frac{\boldsymbol{x}_\alpha - \bar{\boldsymbol{x}}_\alpha}{\varepsilon^2}. \quad (123)$$

Since Eq. (120) is an identity in $\bar{\boldsymbol{x}}_\alpha$ and $\boldsymbol{u}$ that satisfies Eq. (1), the variation (121) should vanish for arbitrary infinitesimal variations $\delta \bar{\boldsymbol{x}}_\alpha$ and $\delta \boldsymbol{u}$ that are compatible with Eq. (1). If Eq. (122) is substituted into Eq. (121), its vanishing means

$$E[(\hat{\boldsymbol{u}} - \boldsymbol{u}) \sum_{\alpha=1}^{N} \boldsymbol{l}_\alpha^\top \delta \bar{\boldsymbol{x}}_\alpha] = \delta \boldsymbol{u}. \quad (124)$$

The infinitesimal variation of Eq. (1) has the form

$$(\nabla_{\mathbf{x}} \bar{F}_\alpha, \delta \bar{\boldsymbol{x}}_\alpha) + (\nabla_{\mathbf{u}} \bar{F}_\alpha, \delta \boldsymbol{u}) = 0, \quad (125)$$

where the overbar means evaluating it at $\boldsymbol{x} = \bar{\boldsymbol{x}}_\alpha$ for the true value $\boldsymbol{u}$. Consider the following particular variations $\delta \bar{\boldsymbol{x}}_\alpha$:

$$\delta \bar{\boldsymbol{x}}_\alpha = -\frac{(\nabla_{\mathbf{x}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \delta \boldsymbol{u}. \quad (126)$$

Evidently, Eq. (125) is satisfied by whatever $\boldsymbol{u}$. Substituting Eq. (126) into Eq. (124), we obtain

$$E[(\hat{\boldsymbol{u}} - \boldsymbol{u}) \sum_{\alpha=1}^{N} \boldsymbol{m}_\alpha^\top] \delta \boldsymbol{u} = -\delta \boldsymbol{u}, \quad (127)$$

where we define the vectors $\boldsymbol{m}_\alpha$ by

$$\boldsymbol{m}_\alpha = \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{x}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \boldsymbol{l}_\alpha. \quad (128)$$

Since Eq. (127) should hold for arbitrary variation $\delta \boldsymbol{u}$, we have

$$E[(\hat{\boldsymbol{u}} - \boldsymbol{u}) \sum_{\alpha=1}^{N} \boldsymbol{m}_\alpha^\top] = -\boldsymbol{I}. \quad (129)$$

Hence, we have

$$E[\begin{pmatrix} \hat{\boldsymbol{u}} - \boldsymbol{u} \\ \sum_{\alpha=1}^{N} \boldsymbol{m}_\alpha \end{pmatrix} \begin{pmatrix} \hat{\boldsymbol{u}} - \boldsymbol{u} \\ \sum_{\alpha=1}^{N} \boldsymbol{m}_\alpha \end{pmatrix}^\top] = \begin{pmatrix} V[\hat{\boldsymbol{u}}] & -\boldsymbol{I} \\ -\boldsymbol{I} & \boldsymbol{M} \end{pmatrix}, \quad (130)$$

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

$$\begin{aligned}\boldsymbol{M} &= E[\Big(\sum_{\alpha=1}^{N} \boldsymbol{m}_\alpha\Big)\Big(\sum_{\beta=1}^{N} \boldsymbol{m}_\beta\Big)^\top] \\ &= \sum_{\alpha,\beta=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{x}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} E[\boldsymbol{l}_\alpha \boldsymbol{l}_\beta] \frac{(\nabla_{\mathbf{x}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \\ &= \frac{1}{\varepsilon^2} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2}. \quad (131)\end{aligned}$$

In the above equation, we use the identity $E[\boldsymbol{l}_\alpha \boldsymbol{l}_\beta^\top] = \delta_{\alpha\beta} \boldsymbol{I}/\varepsilon^4$, which is a consequence of independence of the noise in each datum $\boldsymbol{x}_\alpha$.

Since the inside of the expectation $E[\,\cdot\,]$ on the left-hand side of Eq. (130) is evidently positive semidefinite, so is the right-hand side. Hence, the following is also positive semidefinite:

$$\begin{aligned}&\begin{pmatrix} \boldsymbol{I} & \boldsymbol{M}^{-1} \\ & \boldsymbol{M}^{-1} \end{pmatrix} \begin{pmatrix} V[\hat{\boldsymbol{u}}] & -\boldsymbol{I} \\ -\boldsymbol{I} & \boldsymbol{M} \end{pmatrix} \begin{pmatrix} \boldsymbol{I} & \\ \boldsymbol{M}^{-1} & \boldsymbol{M}^{-1} \end{pmatrix} \\ &= \begin{pmatrix} V[\hat{\boldsymbol{u}}] - \boldsymbol{M}^{-1} & \\ & \boldsymbol{M}^{-1} \end{pmatrix}. \quad (132)\end{aligned}$$

From this, we conclude that

$$V[\hat{\boldsymbol{u}}] \succ \boldsymbol{M}^{-1}. \quad (133)$$

This result is easily generalized to the case where intrinsic constraints exist on the data $\boldsymbol{x}_\alpha$ and the parameter $\boldsymbol{u}$ and the covariance matrix $V[\boldsymbol{x}_\alpha]$ is not full rank. In the general case, we obtain Eq. (9).

B: Linear Approximation of ML

For simplicity, we consider only the case where no intrinsic constraints exist on the data $\boldsymbol{x}_\alpha$ or the parameter $\boldsymbol{u}$ and the noise is identical and isotropic Gaussian. Substituting $\bar{\boldsymbol{x}}_\alpha = \boldsymbol{x}_\alpha - \Delta \boldsymbol{x}_\alpha$ into Eq. (12) and assuming that the noise term $\Delta \boldsymbol{x}_\alpha$ is small, we obtain the linear approximation

$$F_\alpha - (\nabla_{\mathbf{x}} F_\alpha, \Delta \boldsymbol{x}_\alpha) = 0, \quad (134)$$

subject to which we want to minimize $\sum_{\alpha=1}^{N} \|\Delta \boldsymbol{x}_\alpha\|^2$. Introducing Lagrange multipliers λ_α, let

$$L = \frac{1}{2}\sum_{\alpha=1}^{N} \|\Delta \boldsymbol{x}_\alpha\|^2 + \sum_{\alpha=1}^{N} \lambda_\alpha (F_\alpha - (\nabla_{\mathbf{x}} F_\alpha, \Delta \boldsymbol{x}_\alpha)). \tag{135}$$

Taking the derivative of L with respect to $\Delta \boldsymbol{x}_\alpha$ and setting it to $\mathbf{0}$, we have

$$\Delta \boldsymbol{x}_\alpha - \lambda_\alpha \nabla_{\mathbf{x}} F_\alpha = \mathbf{0}. \tag{136}$$

Hence, $\Delta \boldsymbol{x}_\alpha = \lambda_\alpha \nabla_{\mathbf{x}} F_\alpha$. Substitution of this into Eq. (134) yields

$$F_\alpha - (\nabla_{\mathbf{x}} F_\alpha, \lambda_\alpha \nabla_{\mathbf{x}} F_\alpha) = 0, \tag{137}$$

from which we obtain λ_α in the form

$$\lambda_\alpha = \frac{F_\alpha}{\|\nabla_{\mathbf{x}} F_\alpha\|^2}. \tag{138}$$

Thus,

$$\begin{aligned} J &= \sum_{\alpha=1}^{N} \|\Delta \boldsymbol{x}_\alpha\|^2 = \sum_{\alpha=1}^{N} \|\lambda_\alpha \nabla_{\mathbf{x}} F_\alpha\|^2 \\ &= \sum_{\alpha=1}^{N} \frac{F_\alpha^2}{\|\nabla_{\mathbf{x}} F_\alpha\|^4} \|\nabla_{\mathbf{x}} F_\alpha\|^2 = \sum_{\alpha=1}^{N} \frac{F_\alpha^2}{\|\nabla_{\mathbf{x}} F_\alpha\|^2}. \end{aligned} \tag{139}$$

This result can easily be generalized to the case where intrinsic constraints exist on the data $\boldsymbol{x}_\alpha$ and the parameter $\boldsymbol{u}$ and the covariance matrix $V[\boldsymbol{x}_\alpha]$ is not full rank. In the general case, we obtain (15).

C: Covariance Matrix of ML

For simplicity, we consider only the case where no intrinsic constraints exist on the data $\boldsymbol{x}_\alpha$ or the parameter $\boldsymbol{u}$ and the noise is identical and isotropic Gaussian with mean 0 and variance ε^2, so $V[\boldsymbol{x}_\alpha] = \varepsilon^2 \boldsymbol{I}$. Letting $\boldsymbol{x}_\alpha = \bar{\boldsymbol{x}}_\alpha + \Delta \boldsymbol{x}_\alpha$ and replacing $\boldsymbol{u}$ by $\boldsymbol{u} + \Delta \boldsymbol{u}$ in Eq. (15), we can expand J in the form

$$J = \sum_{\alpha=1}^{N} \frac{((\nabla_{\mathbf{x}} \bar{F}_\alpha, \Delta \boldsymbol{x}_\alpha) + (\nabla_{\mathbf{u}} \bar{F}_\alpha, \Delta \boldsymbol{u}))^2}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} + O(\varepsilon^3), \tag{140}$$

where the overbar means evaluating it at $\boldsymbol{x} = \bar{\boldsymbol{x}}_\alpha$ for the true value $\boldsymbol{u}$. Note that replacing $\nabla_{\mathbf{x}} F_\alpha$ by $\nabla_{\mathbf{x}} \bar{F}_\alpha$ by in the denominator does not affect the leading term because the numerator is $O(\varepsilon^2)$; the difference is absorbed into the remainder term $O(\varepsilon^3)$.

If we find $\Delta \boldsymbol{u}$ that minimizes Eq. (140), the ML estimator $\hat{\boldsymbol{u}}$ is given by $\boldsymbol{u} + \Delta \boldsymbol{u}$. Since the first term on the right-hand side of Eq. (140) is quadratic in $\Delta \boldsymbol{u}_\alpha$, the derivative of J with respect to $\Delta \boldsymbol{u}$ is

$$2\sum_{\alpha=1}^{N} \frac{((\nabla_{\mathbf{x}} \bar{F}_\alpha, \Delta \boldsymbol{x}_\alpha) + (\nabla_{\mathbf{u}} \bar{F}_\alpha, \Delta \boldsymbol{u}))\nabla_{\mathbf{u}} \bar{F}_\alpha}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} + O(\varepsilon^2). \tag{141}$$

Letting this be 0, we have

$$\begin{aligned} &\sum_{\alpha=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \Delta \boldsymbol{u} \\ &\quad = -\sum_{\alpha=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{x}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \Delta \boldsymbol{x}_\alpha + O(\varepsilon^2), \end{aligned} \tag{142}$$

from which we obtain

$$\begin{aligned} &\sum_{\alpha=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \Delta \boldsymbol{u} \Delta \boldsymbol{u}^\top \sum_{\beta=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\beta)(\nabla_{\mathbf{u}} \bar{F}_\beta)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\beta\|^2} \\ &= \sum_{\alpha,\beta=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{x}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \Delta \boldsymbol{x}_\alpha \Delta \boldsymbol{x}_\beta^\top \frac{(\nabla_{\mathbf{x}} \bar{F}_\beta)(\nabla_{\mathbf{u}} \bar{F}_\beta)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \\ &\quad + O(\varepsilon^3). \end{aligned} \tag{143}$$

Taking expectation on both sides, we obtain

$$\begin{aligned} &\sum_{\alpha=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} V[\hat{\boldsymbol{u}}] \sum_{\beta=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\beta)(\nabla_{\mathbf{u}} \bar{F}_\beta)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\beta\|^2} \\ &= \sum_{\alpha=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{x}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} \frac{(\nabla_{\mathbf{x}} \bar{F}_\alpha)(\nabla_{\mathbf{u}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} + O(\varepsilon^4) \\ &= \sum_{\alpha=1}^{N} \frac{(\nabla_{\mathbf{u}} \bar{F}_\alpha)(\nabla_{\mathbf{x}} \bar{F}_\alpha)^\top}{\|\nabla_{\mathbf{x}} \bar{F}_\alpha\|^2} + O(\varepsilon^4). \end{aligned} \tag{144}$$

Note that $E[O(\varepsilon^3)] = O(\varepsilon^4)$, because the noise distribution is isotropic and odd noise terms vanish in expectation. The first term in the last expression is the KCR lower bound in this case.

This result can easily be generalized to the case where intrinsic constraints exist on the data $\boldsymbol{x}_\alpha$ and the parameter $\boldsymbol{u}$ and the covariance matrix $V[\boldsymbol{x}_\alpha]$ is not full rank. We conclude that the covariance matrix of the ML estimator agrees with the KCR lower bound except for $O(\varepsilon^4)$.

D: Procedure for the Taubin Method

In most vision applications, the embedded data $\boldsymbol{\xi}_\alpha$, the parameter $\boldsymbol{u}$, and the normalized covariance matrix $V_0[\boldsymbol{\xi}_\alpha]$ are decomposed in the form

$$\begin{aligned} \boldsymbol{\xi}_\alpha &= \begin{pmatrix} \boldsymbol{z}_\alpha \\ C \end{pmatrix}, \quad \boldsymbol{u} = \begin{pmatrix} \boldsymbol{v} \\ a \end{pmatrix}, \\ V_0[\boldsymbol{\xi}_\alpha] &= \begin{pmatrix} V_0[\boldsymbol{z}_\alpha] & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{pmatrix}, \end{aligned} \tag{145}$$

where C and a are constants; see Eqs. (21) and (22) for ellipse fitting and Eqs. (25) and (26) for fundamental matrix computation. Here, $\boldsymbol{z}_\alpha$ and $\boldsymbol{v}$ are $(p-1)$-dimensional vectors, and $V_0[\boldsymbol{z}_\alpha]$ is a $(p-1)\times(p-1)$ normalized covariance matrix of $\boldsymbol{z}_\alpha$; see Eqs. (23) and (27).

So, we compute estimates $\hat{\boldsymbol{v}}$ and $\hat{a}$ of $\boldsymbol{v}$ and a, respectively. Define $(p-1) \times (p-1)$ matrices $\tilde{\boldsymbol{M}}_0$ and $\tilde{\boldsymbol{N}}_0$ by

$$\tilde{\boldsymbol{M}}_0 = \sum_{\alpha=1}^{N} \tilde{\boldsymbol{z}}_\alpha \tilde{\boldsymbol{z}}_\alpha^\top, \qquad \tilde{\boldsymbol{N}}_0 = \sum_{\alpha=1}^{N} V_0[\boldsymbol{z}_\alpha], \tag{146}$$

where

$$\tilde{\boldsymbol{z}}_\alpha = \boldsymbol{z}_\alpha - \bar{\boldsymbol{z}}, \qquad \bar{\boldsymbol{z}} = \frac{1}{N} \sum_{\alpha=1}^{N} \boldsymbol{z}_\alpha. \tag{147}$$

Then, Eq. (45) is decomposed into two equations

$$\tilde{\boldsymbol{M}}_0 \hat{\boldsymbol{v}} = \lambda \tilde{\boldsymbol{N}}_0 \hat{\boldsymbol{v}}, \qquad (\hat{\boldsymbol{v}}, \bar{\boldsymbol{z}}) + C\hat{a} = 0. \tag{148}$$

Compute the $(p-1)$-dimensional unit generalized eigenvector $\hat{\boldsymbol{v}}$ of the first equation for the smallest generalized eigenvalue λ (see, e.g., [12] for the procedure) . The second equation gives $\hat{a}$, and $\hat{\boldsymbol{u}}$ is given by

$$\hat{\boldsymbol{u}} = N[\begin{pmatrix} \hat{\boldsymbol{v}} \\ \hat{a} \end{pmatrix}], \tag{149}$$

where $N[\,\cdot\,]$ denotes normalization to unit norm.

E: Procedure for FNS

The FNS of Chojnacki et al. [7] solves Eq. (103) by the following iterations:

1. Initialize $\hat{\boldsymbol{u}}$, say by LS.
2. Compute the matrix $\hat{\boldsymbol{M}}$ in Eq. (63) and the matrix $\hat{\boldsymbol{L}}$ in Eq. (104), and solve the eigenvalue problem

$$(\hat{\boldsymbol{M}} - \hat{\boldsymbol{L}})\boldsymbol{u} = \lambda \boldsymbol{u}. \tag{150}$$

   Let $\boldsymbol{u}$ be the unit eigenvector for the eigenvalue λ closest to 0.
3. If $\boldsymbol{u} \approx \hat{\boldsymbol{u}}$ except for sign, stop. Else, let $\hat{\boldsymbol{u}} \leftarrow \boldsymbol{u}$ and go back to Step 2.

Later, Chojnacki et al. [9] pointed out that convergence performance improves if we choose in Step 2 not the eigenvalue closest to 0 but the smallest one.

F: Procedure for HEIV

In most vision applications, the embedded data $\boldsymbol{\xi}_\alpha$, the parameter $\boldsymbol{u}$, and the normalized covariance matrix $V_0[\boldsymbol{\xi}_\alpha]$ are decomposed in the form of Eqs. (145). So, we compute estimates $\hat{\boldsymbol{v}}$ and $\hat{a}$ of $\boldsymbol{v}$ and a, respectively. Define $(p-1) \times (p-1)$ matrices $\tilde{\boldsymbol{M}}$ and $\tilde{\boldsymbol{L}}$ by

$$\tilde{\boldsymbol{M}} = \sum_{\alpha=1}^{N} \frac{\tilde{\boldsymbol{z}}_\alpha \tilde{\boldsymbol{z}}_\alpha^\top}{(\hat{\boldsymbol{v}}, V_0[\boldsymbol{z}_\alpha]\hat{\boldsymbol{v}})}, \qquad \tilde{\boldsymbol{L}} = \sum_{\alpha=1}^{N} \frac{(\hat{\boldsymbol{v}}, \tilde{\boldsymbol{z}}_\alpha)^2 V_0[\boldsymbol{z}_\alpha]}{(\hat{\boldsymbol{v}}, V_0[\boldsymbol{z}_\alpha]\hat{\boldsymbol{v}})^2}, \tag{151}$$

where we put

$$\tilde{\boldsymbol{z}}_\alpha = \boldsymbol{z}_\alpha - \bar{\boldsymbol{z}},$$
$$\bar{\boldsymbol{z}} = \sum_{\alpha=1}^{N} \frac{\boldsymbol{z}_\alpha}{(\hat{\boldsymbol{v}}, V_0[\boldsymbol{z}_\alpha]\hat{\boldsymbol{v}})} \Big/ \sum_{\beta=1}^{N} \frac{1}{(\hat{\boldsymbol{v}}, V_0[\boldsymbol{z}_\beta]\hat{\boldsymbol{v}})}. \tag{152}$$

Then, Eq. (103) is decomposed into the following two equations:

$$\tilde{\boldsymbol{M}}\hat{\boldsymbol{v}} = \tilde{\boldsymbol{L}}\hat{\boldsymbol{v}}, \qquad (\hat{\boldsymbol{v}}, \bar{\boldsymbol{z}}) + C\hat{a} = 0. \tag{153}$$

If determine $\hat{\boldsymbol{v}}$ from the first equation, the second equation determines $\hat{a}$, and the estimate $\hat{\boldsymbol{u}}$ is given in the form of Eq. (149). The HEIV of Leedan and Meer [20] solves the first equation by the following iterations:

1. Initialize $\hat{\boldsymbol{v}}$, say by LS.
2. Compute the matrices $\tilde{\boldsymbol{M}}$ and $\tilde{\boldsymbol{L}}$ in Eq. (151), and solve the generalized eigenvalue problem

$$\tilde{\boldsymbol{M}}\boldsymbol{v} = \lambda \tilde{\boldsymbol{L}}\boldsymbol{v}. \tag{154}$$

   Let $\boldsymbol{v}$ be the unit generalized eigenvector for the generalized eigenvalue λ closest to 1.
3. If $\boldsymbol{v} \approx \hat{\boldsymbol{v}}$ except for sign, return $\hat{\boldsymbol{v}}$ and stop. Else, let $\hat{\boldsymbol{v}} \leftarrow \boldsymbol{v}$ and go back to Step 2.

Leedan and Meer [20] pointed out that choosing in Step 3 not the generalized eigenvalue closest to 1 but the smallest one improves the convergence performance.

G: Gauss-Newton Iterations

Since the gradient $\nabla_{\mathbf{u}} J$ is given by Eq. (102), we can minimize the function J in Eq. (101) by Newton iterations. If we evaluate the Hessian $\nabla_{\mathbf{u}}^2 J$, the increment $\Delta\boldsymbol{u}$ in $\boldsymbol{u}$ is determined by solving

$$(\nabla_{\mathbf{u}}^2 J)\Delta\boldsymbol{u} = -\nabla_{\mathbf{u}} J. \tag{155}$$

Since $\nabla_{\mathbf{u}}^2 J$ is singular (the function J is constant in the direction of $\boldsymbol{u}$), the solution is indeterminate. However, if we use pseudoinverse and compute

$$\Delta\boldsymbol{u} = -(\nabla_{\mathbf{u}}^2 J)^- \nabla_{\mathbf{u}} J, \tag{156}$$

we obtain a solution, which is orthogonal to $\boldsymbol{u}$.

Differentiating Eq. (101) and introducing Gauss-Newton approximation (i.e., ignoring terms that contain $(\boldsymbol{u}, \boldsymbol{\xi}_\alpha)$), we see that the Hessian is nothing but the matrix $\hat{\boldsymbol{M}}$ in Eq. (63) for $\boldsymbol{u} = \hat{\boldsymbol{u}}$. In order to compute pseudoinverse, we enforce $\hat{\boldsymbol{M}}$, which is generally nonsingular, to have eigenvalue 0, using the projection matrix $\boldsymbol{P}_{\hat{\mathbf{u}}} = \boldsymbol{I} - \hat{\boldsymbol{u}}\hat{\boldsymbol{u}}^\top$. The iteration procedure goes as follows [19]:

1. Initialize $\hat{\boldsymbol{u}}$, say by LS.
2. Compute

$$\boldsymbol{u} = N[\hat{\boldsymbol{u}} - (\boldsymbol{P}_{\hat{\mathbf{u}}}\hat{\boldsymbol{M}}\boldsymbol{P}_{\hat{\mathbf{u}}})^-(\hat{\boldsymbol{M}} - \hat{\boldsymbol{L}})\hat{\boldsymbol{u}}]. \tag{157}$$

3. If $\boldsymbol{u} \approx \hat{\boldsymbol{u}}$, return $\hat{\boldsymbol{u}}$ and stop. Else, let $\hat{\boldsymbol{u}} \leftarrow \boldsymbol{u}$ and go back to Step 2.