Memoirs of the Faculty of Engineering, Okayama University, Vol. 45, pp. 10–19, January 2011

Optimal Computation of 3-D Rotation under Inhomogeneous Anisotropic Noise

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

(Received November 15, 2010)

We present a new method for optimally computing the 3-D rotation from two sets of 3-D data. Unlike 2-D data, the noise in 3-D data is inherently inhomogeneous and anisotropic, reflecting the characteristics of the 3-D sensing used. To cope with this, Ohta and Kanatani introduced a technique called "renormalization". Following them, we represent a 3-D rotation in terms of a quaternion and compute an exact maximum likelihood solution using the FNS of Chojnacki et al. As an example, we consider 3-D data obtained by stereo vision and optimally compute the 3-D rotation by analyzing the noise characteristics of stereo reconstruction. We show that the widely used method is not suitable for 3-D data. We confirm that the renormalization of Ohta and Kanatani indeed computes almost an optimal solution and that, although the difference is small, the proposed method can compute an even better solution.

1. INTRODUCTION

The task of autonomous robots to reconstruct the 3-D structure of the scene using stereo vision and simultaneously compute its location in the map of the environment is called SLAM (Simultaneous Localization and Mapping) and is one of the central themes of robotics studies today. One of the fundamental techniques for this is to compute the 3-D motion (translation and rotation) of the robot between two time instances. This information is acquired by computing the 3-D motion of the scene relative to the robot. Translation is easily computed by the time change of the centroid of the 3-D points that the robot is tracking. However, rotation is not so easy to compute, because 3-D data, unlike 2-D data, necessarily have inhomogeneous and anisotropic noise originating from the nature of 3-D sensing. If this fact is ignored, a correct rotation cannot be computed.

Similar problems occur in reconstructing the entire 3-D object shape using 3-D sensing. We need multiple sensors, because each sensor can reconstruct only the part that is visible from it. In order to obtain the entire 3-D shape, we need to integrate multiple 3-D parts reconstructed from different sensors. However, each sensor has different noise characteristics, depending on its type, position and orientation. If this fact is ignored, relative rotations among different parts cannot be correctly computed.

This problem is not limited to computer vision but is universal to all problems involving 3-D sensing, including geodetic science, which concerns measurement of the earth surface from multiple satellite sensor data [4]. Thus, 3-D rotation estimation is an important problem in many engineering applications and has been extensively studied since 1980s [1, 3, 7, 8, 11, 19]. However, almost all proposed algorithms assume homogeneous and isotropic noise. Among them, which are all mathematically equivalent, the simplest formulation may be the use of the singular value decomposition (SVD) [10, 11, 19] (Appendix A).

However, the assumption of homogeneous and isotropic noise is totally unrealistic to 3-D data. For 2-D image positions extracted by image processing operations, it may be natural to assume that the x and y coordinates undergo homogeneous and isotropic noise unless the images are known to have special positional and directional characteristics. However, 3-D data are acquired by 3-D sensing such as stereo vision and laser/ultrasonic range finders. Their accuracy is different in the depth orientation and the direction orthogonal to it, resulting in an inhomogeneous and anisotropic noise distribution.

It is Ohta and Kanatani [18] who first pointed out the inevitable inhomogeneity and anisotropy of the noise in 3-D data and presented an optimal 3-D rotation estimation scheme that takes it into account. They used a technique called *renormalization* (Appendix B), which iteratively removes statistical bias of reweight least squares by doing detailed statistical

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

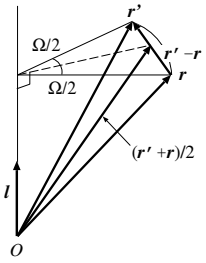

Figure 1: Geometry of 3-D rotation.

error analysis [12]. As a result, a solution statistically equivalent to maximum likelihood (ML) is obtained. However, the solution does not necessarily coincide with the exact ML solution. Later, Chojnacki et al. [2] proposed an iterative scheme, called *FNS* (*Fundamental Numerical Scheme*), similar to renormalization but able to compute an exact ML solution. The same solution can be computed by the method called *HEIV* (*Heteroscedastic Errors in Variable*) of Leedan and Meer [16] and Matei and Meer [17] as well.

In this paper, we adopt, following Ohta and Kanatani [18], the quaternion representation of 3-D rotation and derive a scheme for computing an exact ML solution using the FNS of Chojnacki et al. [2]. Analyzing the uncertainty of 3-D reconstruction by stereo vision, we optimally estimate the 3-D rotation and compare the result with the theoretical accuracy limit called the *KCR lower bound* [12, 13]. It is shown that the widely used method assuming homogeneous and isotropic noise [1, 3, 7, 8, 11, 19] performs very poorly. It is confirmed that the renormalization of Ohta and Kanatani [18] indeed produces almost an optimal solution and that our new method can compute a slightly more accurate solution.

2. QUATERNION REPRESENTATION

If a point $\boldsymbol{r}$ rotates around an axis $\boldsymbol{l}$ (unit vector) by angle Ω (radian) screwwise to $\boldsymbol{r}'$, the geometry of rotation implies the following relationship (Fig. 1):

$$\boldsymbol{r}' - \boldsymbol{r} = 2\tan\frac{\Omega}{2}\boldsymbol{l} \times \frac{\boldsymbol{r}+\boldsymbol{r}'}{2}. \tag{1}$$

This is rewritten as

$$q_0(\boldsymbol{r}' - \boldsymbol{r}) - \boldsymbol{q}_{\boldsymbol{l}} \times (\boldsymbol{r}' + \boldsymbol{r}) = \boldsymbol{0}, \tag{2}$$

where we define

$$q_0 = \cos\frac{\Omega}{2}, \qquad \boldsymbol{q}_{\boldsymbol{l}} = \boldsymbol{l}\sin\frac{\Omega}{2}. \tag{3}$$

This definition implies $q_0^2 + \|\boldsymbol{q}_{\boldsymbol{l}}\|^2 = 1$. Hence, a 3-D rotation is specified by a unit vector

$$\boldsymbol{q} = \begin{pmatrix} q_0 \\ \boldsymbol{q}_{\boldsymbol{l}} \end{pmatrix}, \tag{4}$$

which is known as the *quaternion*[1]. Given a quaternion $\boldsymbol{q}$, the angle Ω and the axis $\boldsymbol{l}$ of the rotation it represents are given by

$$\Omega = 2\cos^{-1} q_0, \qquad \boldsymbol{l} = \mathcal{N}[\boldsymbol{q}_{\boldsymbol{l}}], \tag{5}$$

where $\mathcal{N}[\,\cdot\,]$ denotes normalization into unit norm. In the following, we define the product $\boldsymbol{a} \times \boldsymbol{T}$ of a vector $\boldsymbol{a}$ and a matrix $\boldsymbol{T}$ as the matrix whose three columns are the vector products of $\boldsymbol{a}$ and the respective columns of $\boldsymbol{T}$. From this definition, we see that for a vector $\boldsymbol{a} = (a_i)$ and the unit matrix $\boldsymbol{I}$

$$\boldsymbol{a} \times \boldsymbol{I} = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix}, \tag{6}$$

which is an antisymmetric matrix. It is easy to see the identities $(\boldsymbol{a} \times \boldsymbol{I})\boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$ and $(\boldsymbol{a} \times \boldsymbol{I})\boldsymbol{T} = \boldsymbol{a} \times \boldsymbol{T}$ for any vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ and any matrix $\boldsymbol{T}$. Hereafter, we abbreviate $\boldsymbol{T}(\boldsymbol{a} \times \boldsymbol{I})^\top$ to $\boldsymbol{T} \times \boldsymbol{a}$.

3. OPTIMAL ESTIMATION

Suppose we have measurement data of 3-D positions $\boldsymbol{r}_\alpha$ before rotation and their positions $\boldsymbol{r}'_\alpha$ after rotation, $\alpha = 1, ..., N$. Measurement entails uncertainty to some extent. Modeling it by Gaussian noise, we assume that the covariance matrices of $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$ have the form of $\epsilon^2 V_0[\boldsymbol{r}_\alpha]$ and $\epsilon^2 V_0[\boldsymbol{r}'_\alpha]$, respectively, where ϵ, which we call the *noise level*, represents the magnitude of the noise, while $V_0[\boldsymbol{r}_\alpha]$ and $V_0[\boldsymbol{r}'_\alpha]$, which we call the *normalized covariance matrices*, describe the directional characteristics of the noise distribution. Optimal estimation in the sense of maximum likelihood (ML) is to minimize the *Mahalanobis distance* (the multiplier 1/2 is merely for convenience)

$$J = \frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{r}_\alpha - \bar{\boldsymbol{r}}_\alpha, V_0[\boldsymbol{r}_\alpha]^{-1}(\boldsymbol{r}_\alpha - \bar{\boldsymbol{r}}_\alpha)) + \frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{r}'_\alpha - \bar{\boldsymbol{r}}'_\alpha, V_0[\boldsymbol{r}'_\alpha]^{-1}(\boldsymbol{r}'_\alpha - \bar{\boldsymbol{r}}'_\alpha)), \tag{7}$$

with respect to $\bar{\boldsymbol{r}}_\alpha$, $\bar{\boldsymbol{r}}'_\alpha$ subject to

$$q_0(\bar{\boldsymbol{r}}'_\alpha - \bar{\boldsymbol{r}}_\alpha) - \boldsymbol{q}_{\boldsymbol{l}} \times (\bar{\boldsymbol{r}}'_\alpha + \bar{\boldsymbol{r}}_\alpha) = \boldsymbol{0}, \tag{8}$$

for some q_0 and $\boldsymbol{q}_{\boldsymbol{l}}$. Throughout this paper, we denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ by $(\boldsymbol{a}, \boldsymbol{b})$. If we let $V_0[\boldsymbol{r}_\alpha] = V_0[\boldsymbol{r}'_\alpha] = \boldsymbol{I}$, the above formulation reduces to the case of homogeneous isotropic noise, which has been extensively studied in the past [1, 7, 8, 11, 19]. If that is the case, we may regard $\boldsymbol{r}_\alpha$ as noiseless and only $\boldsymbol{r}'_\alpha$ as noisy, or $\boldsymbol{r}'_\alpha$ as noiseless

[1]Mathematically, $\boldsymbol{q}$ is called a "quaternion" when associated with its algebra, i.e., the rule of composition [9]. However, the quaternion algebra does not play any role in this paper.

and $\boldsymbol{r}_\alpha$ as noisy; the solutions are the same [5]. If the noise is inhomogeneous and anisotropic, however, the noise characteristics before and after the rotation both affect the solution. Introducing Lagrangian multipliers $\boldsymbol{\lambda}_\alpha$ for the constraint of Eq. (8), we consider

$$\begin{aligned}\tilde{J} = &\frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{r}_\alpha - \bar{\boldsymbol{r}}_\alpha, V_0[\boldsymbol{r}_\alpha]^{-1}(\boldsymbol{r}_\alpha - \bar{\boldsymbol{r}}_\alpha))\\ &+\frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{r}'_\alpha - \bar{\boldsymbol{r}}'_\alpha, V_0[\boldsymbol{r}'_\alpha]^{-1}(\boldsymbol{r}'_\alpha - \bar{\boldsymbol{r}}'_\alpha))\\ &-\sum_{\alpha=1}^{N}(\boldsymbol{\lambda}_\alpha, q_0(\bar{\boldsymbol{r}}'_\alpha - \bar{\boldsymbol{r}}_\alpha) - \boldsymbol{q_l}\times(\bar{\boldsymbol{r}}'_\alpha + \bar{\boldsymbol{r}}_\alpha)). \quad (9)\end{aligned}$$

Using the identities $(\boldsymbol{\lambda}_\alpha, \boldsymbol{q_l}\times\bar{\boldsymbol{r}}_\alpha) = -(\boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha, \bar{\boldsymbol{r}}_\alpha)$ and $(\boldsymbol{\lambda}_\alpha, \boldsymbol{q_l}\times\bar{\boldsymbol{r}}'_\alpha) = -(\boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha, \bar{\boldsymbol{r}}'_\alpha)$, we can obtain the derivatives of $\tilde{J}$ with respect to $\bar{\boldsymbol{r}}_\alpha$ and $\bar{\boldsymbol{r}}'_\alpha$ in the form

$$\begin{aligned}\nabla_{\bar{\boldsymbol{r}}_\alpha}\tilde{J} &= -V_0[\boldsymbol{r}_\alpha]^{-1}(\boldsymbol{r}_\alpha - \bar{\boldsymbol{r}}_\alpha) + q_0\boldsymbol{\lambda}_\alpha - \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha,\\ \nabla_{\bar{\boldsymbol{r}}'_\alpha}\tilde{J} &= -V_0[\boldsymbol{r}'_\alpha]^{-1}(\boldsymbol{r}'_\alpha - \bar{\boldsymbol{r}}'_\alpha) - q_0\boldsymbol{\lambda}_\alpha - \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha.\end{aligned} \quad (10)$$

Setting these to $\boldsymbol{0}$, we obtain

$$\begin{aligned}\bar{\boldsymbol{r}}_\alpha &= \boldsymbol{r}_\alpha - V_0[\boldsymbol{r}_\alpha](q_0\boldsymbol{\lambda}_\alpha - \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha),\\ \bar{\boldsymbol{r}}'_\alpha &= \boldsymbol{r}'_\alpha + V_0[\boldsymbol{r}'_\alpha](q_0\boldsymbol{\lambda}_\alpha + \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha).\end{aligned} \quad (11)$$

Substitution of these into Eq. (8) yields

$$\begin{aligned}&q_0(\boldsymbol{r}'_\alpha - \boldsymbol{r}_\alpha) + q_0^2(V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])\boldsymbol{\lambda}_\alpha\\ &+q_0(V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])(\boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha) - \boldsymbol{q_l}\times(\boldsymbol{r}'_\alpha + \boldsymbol{r}_\alpha)\\ &-\boldsymbol{q_l}\times(q_0(V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{\lambda}_\alpha)\\ &-\boldsymbol{q_l}\times(V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])(\boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha) = \boldsymbol{0},\end{aligned} \quad (12)$$

which is rewritten as

$$-q_0(\boldsymbol{r}'_\alpha - \boldsymbol{r}_\alpha) + \boldsymbol{q_l}\times(\boldsymbol{r}'_\alpha + \boldsymbol{r}_\alpha) = \boldsymbol{V}_\alpha\boldsymbol{\lambda}_\alpha, \quad (13)$$

where we define the matrix $\boldsymbol{V}_\alpha$ by

$$\begin{aligned}\boldsymbol{V}_\alpha = &q_0^2(V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])\\ &-2q_0\mathcal{S}[\boldsymbol{q_l}\times(V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])]\\ &+\boldsymbol{q_l}\times(V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])\times\boldsymbol{q_l}.\end{aligned} \quad (14)$$

The operator $\mathcal{S}[\,\cdot\,]$ designates symmetrization ($\mathcal{S}[\boldsymbol{A}] = (\boldsymbol{A} + \boldsymbol{A}^\top)/2$). Letting $\boldsymbol{W}_\alpha = \boldsymbol{V}_\alpha^{-1}$, we obtain $\boldsymbol{\lambda}_\alpha$ from Eq. (13) in the form

$$\begin{aligned}\boldsymbol{\lambda}_\alpha &= -\boldsymbol{W}_\alpha\begin{pmatrix}\boldsymbol{r}'_\alpha - \boldsymbol{r}_\alpha & (\boldsymbol{r}'_\alpha + \boldsymbol{r}_\alpha)\times\boldsymbol{I}\end{pmatrix}\begin{pmatrix}q_0\\ \boldsymbol{q_l}\end{pmatrix}\\ &= \boldsymbol{W}_\alpha\boldsymbol{X}_\alpha\boldsymbol{q},\end{aligned} \quad (15)$$

where $\boldsymbol{q}$ is the quaternion defined in Eq. (4), and $\boldsymbol{X}_\alpha$ is a 3×4 matrix in the form

$$\boldsymbol{X}_\alpha = \begin{pmatrix}\boldsymbol{r}'_\alpha - \boldsymbol{r}_\alpha & (\boldsymbol{r}'_\alpha + \boldsymbol{r}_\alpha)\times\boldsymbol{I}.\end{pmatrix} \quad (16)$$

Substituting Eqs. (11) into Eq. (7), we see that

$$\begin{aligned}J = &\frac{1}{2}\sum_{\alpha=1}^{N}\Big((V_0[\boldsymbol{r}_\alpha](q_0\boldsymbol{\lambda}_\alpha - \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha), q_0\boldsymbol{\lambda}_\alpha - \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha)\\ &+(V_0[\boldsymbol{r}'_\alpha](q_0\boldsymbol{\lambda}_\alpha + \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha), q_0\boldsymbol{\lambda}_\alpha + \boldsymbol{q_l}\times\boldsymbol{\lambda}_\alpha)\Big)\\ = &\frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{\lambda}_\alpha, \boldsymbol{V}_\alpha\boldsymbol{\lambda}_\alpha).\end{aligned} \quad (17)$$

After substitution of Eq. (15), we obtain

$$\begin{aligned}J &= \frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{W}_\alpha\boldsymbol{X}_\alpha\boldsymbol{q}, \boldsymbol{V}_\alpha\boldsymbol{W}_\alpha\boldsymbol{X}_\alpha\boldsymbol{q})\\ &= \frac{1}{2}\sum_{\alpha=1}^{N}(\boldsymbol{q}, \boldsymbol{X}_\alpha^\top\boldsymbol{W}_\alpha\boldsymbol{W}_\alpha^{-1}\boldsymbol{W}_\alpha\boldsymbol{X}_\alpha\boldsymbol{q})\\ &= \frac{1}{2}(\boldsymbol{q}, \boldsymbol{M}\boldsymbol{q}),\end{aligned} \quad (18)$$

where we have defined the 4×4 matrix $\boldsymbol{M}$ by

$$\boldsymbol{M} = \sum_{\alpha=1}^{N}\boldsymbol{X}_\alpha^\top\boldsymbol{W}_\alpha\boldsymbol{X}_\alpha. \quad (19)$$

This is the formulation introduced by Ohta and Kanatani [18].

4. OPTIMIZATION PROCEDURE

For minimizing Eq. (18), Ohta and Kanatani [18] used a technique called *renormalization*, which iteratively removes statistical bias of reweight least squares by doing detailed statistical error analysis [12]. As a result, a solution statistically equivalent to ML is obtained, but it does not necessarily coincide with the exact ML solution. Here, we directly minimize Eq. (18). Differentiation of Eq. (18) with respect to q_κ, $\kappa = 0, 1, 2, 3$, gives

$$\frac{\partial J}{\partial q_\kappa} = \sum_{\lambda=0}^{3}M_{\kappa\lambda}q_\lambda + \frac{1}{2}(\boldsymbol{q}, \frac{\partial\boldsymbol{M}}{\partial q_\kappa}\boldsymbol{q}), \quad (20)$$

where $M_{\kappa\lambda}$ is the $(\kappa\lambda)$ element of $\boldsymbol{M}$. From Eq. (19), the derivative of the matrix $\boldsymbol{M}$ is

$$\frac{\partial\boldsymbol{M}}{\partial q_\kappa} = \sum_{\alpha=1}^{N}\boldsymbol{X}_\alpha^\top\frac{\partial\boldsymbol{W}_\alpha}{\partial q_\kappa}\boldsymbol{X}_\alpha. \quad (21)$$

First, we evaluate $\partial\boldsymbol{W}_\alpha/\partial q_\kappa$. The matrix $\boldsymbol{W}_\alpha$ is defined by $\boldsymbol{W}_\alpha = \boldsymbol{V}_\alpha^{-1}$. Differentiating $\boldsymbol{V}_\alpha\boldsymbol{W}_\alpha = \boldsymbol{I}$ with respect to q_κ on both sides, we have

$$\frac{\partial\boldsymbol{V}_\alpha}{\partial q_\kappa}\boldsymbol{W}_\alpha + \boldsymbol{V}_\alpha\frac{\partial\boldsymbol{W}_\alpha}{\partial q_\kappa} = \boldsymbol{O}, \quad (22)$$

from which we obtain $\partial\boldsymbol{W}_\alpha/\partial q_\kappa$ in the form

$$\frac{\partial\boldsymbol{W}_\alpha}{\partial q_\kappa} = -\boldsymbol{V}_\alpha^{-1}\frac{\partial\boldsymbol{V}_\alpha}{\partial q_\kappa}\boldsymbol{W}_\alpha = -\boldsymbol{W}_\alpha\frac{\partial\boldsymbol{V}_\alpha}{\partial q_\kappa}\boldsymbol{W}_\alpha. \quad (23)$$

Hence, $(\boldsymbol{q}, \partial \boldsymbol{M}/\partial q_\kappa \boldsymbol{q})$ is expressed from Eq. (21) in the form

$$\begin{aligned}(\boldsymbol{q}, \frac{\partial \boldsymbol{M}}{\partial q_\kappa}\boldsymbol{q}) &= -(\boldsymbol{q}, \sum_{\alpha=1}^{N} \boldsymbol{X}_\alpha^\top \boldsymbol{W}_\alpha \frac{\partial \boldsymbol{V}_\alpha}{\partial q_\kappa} \boldsymbol{W}_\alpha \boldsymbol{X}_\alpha \boldsymbol{q}) \\ &= -\sum_{\alpha=1}^{N} (\boldsymbol{W}_\alpha \boldsymbol{X}_\alpha \boldsymbol{q}, \frac{\partial \boldsymbol{V}_\alpha}{\partial q_\kappa} \boldsymbol{W}_\alpha \boldsymbol{X}_\alpha \boldsymbol{q}) \\ &= -\sum_{\alpha=1}^{N} (\boldsymbol{p}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial q_\kappa} \boldsymbol{p}_\alpha), \end{aligned} \tag{24}$$

where we put

$$\boldsymbol{p}_\alpha = \boldsymbol{W}_\alpha \boldsymbol{X}_\alpha \boldsymbol{q}. \tag{25}$$

Next, we evaluate $\partial \boldsymbol{V}_\alpha/\partial q_\kappa$, $\kappa = 0, 1, 2, 3$. From Eq. (14), we obtain

$$\begin{aligned}\frac{\partial \boldsymbol{V}_\alpha}{\partial q_0} &= 2q_0(V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \\ &\quad -2\mathcal{S}[\boldsymbol{q_l} \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])] \\ \frac{\partial \boldsymbol{V}_\alpha}{\partial q_1} &= -2q_0\mathcal{S}[\boldsymbol{i} \times (V_0[\boldsymbol{r}_\alpha] - V_0[\boldsymbol{r}'_\alpha])] \\ &\quad +2\mathcal{S}[\boldsymbol{i} \times (V_0[\boldsymbol{r}_\alpha] + V_0[\boldsymbol{r}'_\alpha]) \times \boldsymbol{q_l}] \\ \frac{\partial \boldsymbol{V}_\alpha}{\partial q_2} &= -2q_0\mathcal{S}[\boldsymbol{j} \times (V_0[\boldsymbol{r}_\alpha] - V_0[\boldsymbol{r}'_\alpha])] \\ &\quad +2\mathcal{S}[\boldsymbol{j} \times (V_0[\boldsymbol{r}_\alpha] + V_0[\boldsymbol{r}'_\alpha]) \times \boldsymbol{q_l}] \\ \frac{\partial \boldsymbol{V}_\alpha}{\partial q_3} &= -2q_0\mathcal{S}[\boldsymbol{k} \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])] \\ &\quad +2\mathcal{S}[\boldsymbol{k} \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{q_l}], \end{aligned} \tag{26}$$

where we put $\boldsymbol{i} = (1,0,0)^\top$, $\boldsymbol{j} = (0,1,0)^\top$, and $\boldsymbol{k} = (0,0,1)^\top$. Thus, we can express $(\boldsymbol{p}_\alpha, \partial \boldsymbol{V}_\alpha/\partial q_\kappa \boldsymbol{p}_\alpha)$ as follows:

$$\begin{aligned}(\boldsymbol{p}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial q_0}\boldsymbol{p}_\alpha) &= 2q_0(\boldsymbol{p}_\alpha, (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha) \\ &\quad +2(\boldsymbol{q_l}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha), \\ (\boldsymbol{p}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial q_1}\boldsymbol{p}_\alpha) &= 2q_0(\boldsymbol{i}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha) \\ &\quad +2(\boldsymbol{i}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{p}_\alpha)\boldsymbol{q_l}), \\ (\boldsymbol{p}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial q_2}\boldsymbol{p}_\alpha) &= 2q_0(\boldsymbol{j}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha) \\ &\quad +2(\boldsymbol{j}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{p}_\alpha)\boldsymbol{q_l}), \\ (\boldsymbol{p}_\alpha, \frac{\partial \boldsymbol{V}_\alpha}{\partial q_3}\boldsymbol{p}_\alpha) &= 2q_0(\boldsymbol{k}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha) \\ &\quad +2(\boldsymbol{k}, \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{p}_\alpha)\boldsymbol{q_l}). \end{aligned} \tag{27}$$

From these, the term $(1/2)(\boldsymbol{q}, \partial \boldsymbol{M}/\partial q_\kappa \boldsymbol{q})$ in Eq. (24) is expressed as

$$\begin{aligned}\frac{1}{2}(\boldsymbol{q}, \frac{\partial \boldsymbol{M}}{\partial q_0}\boldsymbol{q}) &= -q_0 a - (\boldsymbol{q_l}, \boldsymbol{b}), \\ \frac{1}{2}(\boldsymbol{q}, \frac{\partial \boldsymbol{M}}{\partial q_1}\boldsymbol{q}) &= -q_0(\boldsymbol{i}, \boldsymbol{b}) - (\boldsymbol{i}, \boldsymbol{C}\boldsymbol{q_l}) \\ \frac{1}{2}(\boldsymbol{q}, \frac{\partial \boldsymbol{M}}{\partial q_2}\boldsymbol{q}) &= -q_0(\boldsymbol{j}, \boldsymbol{b}) - (\boldsymbol{j}, \boldsymbol{C}\boldsymbol{q_l}), \\ \frac{1}{2}(\boldsymbol{q}, \frac{\partial \boldsymbol{M}}{\partial q_3}\boldsymbol{q}) &= -q_0(\boldsymbol{k}, \boldsymbol{b}) - (\boldsymbol{k}, \boldsymbol{C}\boldsymbol{q_l}), \end{aligned} \tag{28}$$

where we define the scalar a, the vector $\boldsymbol{b}$, and the matrix $\boldsymbol{C}$ as follows:

$$\begin{aligned} a &= \sum_{\alpha=1}^{N} (\boldsymbol{p}_\alpha, (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha), \\ \boldsymbol{b} &= \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha, \\ \boldsymbol{C} &= \sum_{\alpha=1}^{N} \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{p}_\alpha. \end{aligned} \tag{29}$$

Thus, the vector consisting of $(1/2)(\boldsymbol{q}, \partial \boldsymbol{M}/\partial q_\kappa \boldsymbol{q})$, $\kappa = 0, 1, 2, 3$, is given by

$$\begin{aligned} &-\begin{pmatrix} q_0 a + (\boldsymbol{q_l}, \boldsymbol{b}) \\ q_0(\boldsymbol{i}, \boldsymbol{b}) + (\boldsymbol{i}, \boldsymbol{C}\boldsymbol{q_l}) \\ q_0(\boldsymbol{j}, \boldsymbol{b}) + (\boldsymbol{j}, \boldsymbol{C}\boldsymbol{q_l}) \\ q_0(\boldsymbol{k}, \boldsymbol{b}) + (\boldsymbol{k}, \boldsymbol{C}\boldsymbol{q_l}) \end{pmatrix} = -\begin{pmatrix} q_0 a + (\boldsymbol{q_l}, \boldsymbol{b}) \\ q_0 \boldsymbol{b} + \boldsymbol{C}\boldsymbol{q_l} \end{pmatrix} \\ &= -\begin{pmatrix} a & \boldsymbol{b}^\top \\ \boldsymbol{b} & \boldsymbol{C} \end{pmatrix} \boldsymbol{q} = -\boldsymbol{L}\boldsymbol{q}, \end{aligned} \tag{30}$$

where $\boldsymbol{L}$ is the following 4×4 matrix:

$$\begin{aligned} \boldsymbol{L} = \sum_{\alpha=1}^{N} \begin{pmatrix} (\boldsymbol{p}_\alpha, (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha) & \\ \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha & \\ (\boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])\boldsymbol{p}_\alpha)^\top & \\ \boldsymbol{p}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{p}_\alpha & \end{pmatrix}. \end{aligned} \tag{31}$$

Now, Eq. (20) can be expressed as a vector equation

$$\nabla_{\boldsymbol{q}} J = \boldsymbol{M}\boldsymbol{q} - \boldsymbol{L}\boldsymbol{q}. \tag{32}$$

5. FNS PROCEDURE

Our task is to compute the unit vector $\boldsymbol{q}$ that makes Eq. (32) $\boldsymbol{0}$, for which we can use the FNS of Chojnacki et al. [2]. The FNS procedure goes as follows:

1. Compute the matrices $\boldsymbol{X}_\alpha$ in Eq. (16) from the positions $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$ before and after the rotation, and provide an initial guess of $\boldsymbol{q}$.
2. Compute the matrices $\boldsymbol{V}_\alpha$ in Eq. (14) and $\boldsymbol{W}_\alpha = \boldsymbol{V}_\alpha^{-1}$.
3. Compute the matrix $\boldsymbol{M}$ in Eq. (19), the vectors $\boldsymbol{p}_\alpha$ in Eq. (25), and the matrix $\boldsymbol{L}$ in Eq. (31).
4. Solve the eigenvalue problem

$$(\boldsymbol{M} - \boldsymbol{L})\boldsymbol{q}' = \lambda \boldsymbol{q}', \tag{33}$$

and compute the unit eigenvector $\boldsymbol{q}'$ corresponding to the smallest eigenvalue λ.

5. If $\boldsymbol{q}' \approx \pm\boldsymbol{q}$, return $\boldsymbol{q}'$ and stop. Else, let $\boldsymbol{q} \leftarrow \boldsymbol{q}'$, and go back to Step 2.

We need an an initial guess of $\boldsymbol{q}$. The simplest choice is, as done by Ohta and Kanatani [18], the use of the unit eigenvector $\boldsymbol{q}$ of the matrix

$$\boldsymbol{M}_0 = \sum_{\alpha=1}^{N} \boldsymbol{X}_\alpha^\top \boldsymbol{X}_\alpha \tag{34}$$

for the smallest eigenvalue.

6. COVARIANCE MATRIX EVALUATION

The above algorithm involves the normalized covariance matrices $V_0[\boldsymbol{r}_\alpha]$ and $V_0[\boldsymbol{r}'_\alpha]$ that characterize the distributions of noise in $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$. The noise characteristics depend on what kind of 3-D sensing is used for measuring $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$. Here, we consider stereo vision. We fix an XYZ world coordinate system and regard the reference camera position as placed at the coordinate origin O with the optical axis aligned to the Z-axis. The image xy coordinate system is defined in such a way that its origin o is at the principal point (the intersection with the optical axis) and the x- and y-axis are parallel to the X- and Y-axis of the world coordinate system, respectively. Then, the camera is rotated around the world coordinate origin O by $\boldsymbol{R}$ (rotation matrix) and translated by $\boldsymbol{t}$ from the reference position. We call $\{\boldsymbol{R}, \boldsymbol{t}\}$ the *motion parameters* of the camera. The camera imaging geometry is modeled by perspective projection with focal length f, projecting a 3-D point onto a 2-D point (x, y) by the following relationship [6]:

$$\boldsymbol{x} \simeq \boldsymbol{P}\boldsymbol{X}, \qquad \boldsymbol{x} \equiv \begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix}, \qquad \boldsymbol{X} \equiv \begin{pmatrix} \boldsymbol{r} \\ 1 \end{pmatrix}. \tag{35}$$

The symbol $\simeq$ means equality up to a nonzero constant multiplier, and f_0 is a scale constant of approximately the image size for stabilizing finite length computation. The 3×4 projection matrix $\boldsymbol{P}$ is given by

$$\boldsymbol{P} = \begin{pmatrix} f/f_0 & 0 & 0 \\ 0 & f/f_0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{R}^\top & -\boldsymbol{R}^\top \boldsymbol{t} \end{pmatrix}, \tag{36}$$

where the aspect ratio is assumed to be 1 with no image skews, or so corrected by prior calibration.

We consider two cameras with motion parameters $\{\boldsymbol{R}, \boldsymbol{t}\}$ and $\{\boldsymbol{R}', \boldsymbol{t}'\}$ with focal lengths f and f', respectively. Let $\boldsymbol{P}$ and $\boldsymbol{P}'$ be the projection matrices of the respective cameras, and $\boldsymbol{x}$ and $\boldsymbol{x}'$ the images of a point in 3-D observed by the respective cameras. Image processing for correspondence detection entails uncertainty to some extent, and we model it by independent isotropic Gaussian noise of mean $\boldsymbol{0}$ and standard deviation σ (pixels). Due to noise, the detected points $\boldsymbol{x}$ and $\boldsymbol{x}'$ do not exactly satisfy the epipolar constraint (Appendix C), so we correct $\boldsymbol{x}$ and $\boldsymbol{x}'$, respectively, to $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}'$ that exactly satisfy the epipolar constraint in an optimal manner (Appendix D). From the corrected positions $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}'$, the corresponding 3-D position $\hat{\boldsymbol{r}}$ is uniquely determined (Appendix C).

Note that although the noise in $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ is assumed to be independent, the noise in the corrected positions $\hat{\boldsymbol{x}}_\alpha$ and $\hat{\boldsymbol{x}}'_\alpha$ is no longer independent [12]. The normalized covariance matrices $V_0[\hat{\boldsymbol{x}}]$ and $V_0[\hat{\boldsymbol{x}}']$ and the normalized correlation matrices $V_0[\hat{\boldsymbol{x}}, \hat{\boldsymbol{x}}']$ and $V_0[\hat{\boldsymbol{x}}', \hat{\boldsymbol{x}}]$ are given as follows [12, 15]:

$$\begin{aligned} V_0[\hat{\boldsymbol{x}}] &= \frac{1}{f_0^2}\Big(\boldsymbol{P}_{\boldsymbol{k}} - \frac{(\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}\hat{\boldsymbol{x}}')(\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}\hat{\boldsymbol{x}}')^\top}{\|\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}\hat{\boldsymbol{x}}'\|^2 + \|\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}^\top\hat{\boldsymbol{x}}\|^2}\Big), \\ V_0[\hat{\boldsymbol{x}}'] &= \frac{1}{f_0^2}\Big(\boldsymbol{P}_{\boldsymbol{k}} - \frac{(\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}^\top\hat{\boldsymbol{x}})(\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}^\top\hat{\boldsymbol{x}})^\top}{\|\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}\hat{\boldsymbol{x}}'\|^2 + \|\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}^\top\hat{\boldsymbol{x}}\|^2}\Big), \\ V_0[\hat{\boldsymbol{x}}, \hat{\boldsymbol{x}}'] &= \frac{1}{f_0^2}\Big(-\frac{(\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}\hat{\boldsymbol{x}}')(\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}^\top\hat{\boldsymbol{x}})^\top}{\|\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}\hat{\boldsymbol{x}}'\|^2 + \|\boldsymbol{P}_{\boldsymbol{k}}\boldsymbol{F}^\top\hat{\boldsymbol{x}}\|^2}\Big) \\ &= V_0[\hat{\boldsymbol{x}}', \hat{\boldsymbol{x}}]^\top. \end{aligned} \tag{37}$$

Here, we define $\boldsymbol{P}_{\boldsymbol{k}} \equiv \mathrm{diag}(1, 1, 0)$.

Since the vector $\hat{\boldsymbol{X}}$ reconstructed from $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}'$ satisfies the projection relationship in Eq. (35), vectors $\hat{\boldsymbol{x}}$ and $\boldsymbol{P}\hat{\boldsymbol{X}}$ are parallel, and so are $\hat{\boldsymbol{x}}'$ and $\boldsymbol{P}'\hat{\boldsymbol{X}}$. Thus, we have

$$\hat{\boldsymbol{x}} \times \boldsymbol{P}\hat{\boldsymbol{X}} = \boldsymbol{0}, \qquad \hat{\boldsymbol{x}}' \times \boldsymbol{P}'\hat{\boldsymbol{X}} = \boldsymbol{0} \tag{38}$$

It follows that if the noise in $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}'$ is $\Delta\hat{\boldsymbol{x}}$ and $\Delta\hat{\boldsymbol{x}}'$, respectively, the noise $\Delta\hat{\boldsymbol{X}}$ in $\hat{\boldsymbol{X}}$ satisfies to a first approximation

$$\begin{aligned} \Delta\hat{\boldsymbol{x}} \times \boldsymbol{P}\hat{\boldsymbol{X}} + \hat{\boldsymbol{x}} \times \boldsymbol{P}\Delta\hat{\boldsymbol{X}} &= \boldsymbol{0}, \\ \Delta\hat{\boldsymbol{x}}' \times \boldsymbol{P}'\hat{\boldsymbol{X}}' + \hat{\boldsymbol{x}}' \times \boldsymbol{P}'\Delta\hat{\boldsymbol{X}} &= \boldsymbol{0}. \end{aligned} \tag{39}$$

These are combined into one equation in the form

$$\begin{pmatrix} \hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}} \\ \hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}' \end{pmatrix} \Delta\hat{\boldsymbol{r}} = \begin{pmatrix} (\boldsymbol{P}\hat{\boldsymbol{X}}) \times \boldsymbol{I} & \boldsymbol{O} \\ \boldsymbol{O} & (\boldsymbol{P}'\hat{\boldsymbol{X}}) \times \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \Delta\hat{\boldsymbol{x}} \\ \Delta\hat{\boldsymbol{x}}' \end{pmatrix}, \tag{40}$$

where $\tilde{\boldsymbol{P}}$ and $\tilde{\boldsymbol{P}}'$ are the left 3×3 submatrices of the 3×4 projection matrices $\boldsymbol{P}$ and $\boldsymbol{P}'$, respectively. Multiplying both sides by the transpose of the left-hand side from left, we obtain

$$\begin{aligned} &\Big((\hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}})^\top(\hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}}) + (\hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}')^\top(\hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}')\Big)\Delta\hat{\boldsymbol{r}} \\ &= \Big((\hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}})^\top((\boldsymbol{P}\hat{\boldsymbol{X}}) \times \boldsymbol{I}) \\ &\qquad \hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}')^\top((\boldsymbol{P}'\hat{\boldsymbol{X}}) \times \boldsymbol{I})\Big) \begin{pmatrix} \Delta\hat{\boldsymbol{x}} \\ \Delta\hat{\boldsymbol{x}}' \end{pmatrix}. \end{aligned} \tag{41}$$

The following identities hold [12]:

$$(\hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}})^\top(\hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}}) = \tilde{\boldsymbol{P}}^\top(\hat{\boldsymbol{x}} \times \boldsymbol{I})^\top(\hat{\boldsymbol{x}} \times \boldsymbol{I})\tilde{\boldsymbol{P}}$$

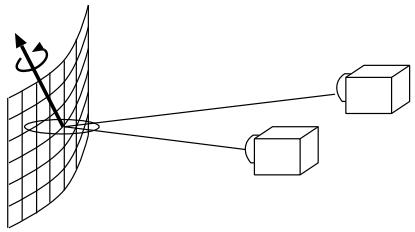

Figure 2: 3-D measurement of a grid point by stereo vision and its uncertainty ellipsoid.

$$
\begin{aligned}
&= \|\hat{\boldsymbol{x}}\|^2 \tilde{\boldsymbol{P}}^\top \boldsymbol{P}_{\mathcal{N}[\hat{\boldsymbol{x}}]} \tilde{\boldsymbol{P}}, \\
(\hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}')^\top (\hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}') &= \tilde{\boldsymbol{P}}'^\top (\hat{\boldsymbol{x}}' \times \boldsymbol{I})^\top (\hat{\boldsymbol{x}}' \times \boldsymbol{I}) \tilde{\boldsymbol{P}}' \\
&= \|\hat{\boldsymbol{x}}'\|^2 \tilde{\boldsymbol{P}}'^\top \boldsymbol{P}_{\mathcal{N}[\hat{\boldsymbol{x}}']} \tilde{\boldsymbol{P}}'. \quad (42)
\end{aligned}
$$

Here, we define

$$
\boldsymbol{P}_{\mathcal{N}[\hat{\boldsymbol{x}}]} \equiv \boldsymbol{I} - \mathcal{N}[\hat{\boldsymbol{x}}]\mathcal{N}[\hat{\boldsymbol{x}}]^\top, \quad \boldsymbol{P}_{\mathcal{N}[\hat{\boldsymbol{x}}']} \equiv \boldsymbol{I} - \mathcal{N}[\hat{\boldsymbol{x}}']\mathcal{N}[\hat{\boldsymbol{x}}']^\top. \tag{43}
$$

Similarly, we have

$$
\begin{aligned}
(\hat{\boldsymbol{x}} \times \tilde{\boldsymbol{P}})^\top ((\boldsymbol{P}\hat{\boldsymbol{X}}) \times \boldsymbol{I}) &= \tilde{\boldsymbol{P}}^\top \Big((\hat{\boldsymbol{x}}, \boldsymbol{P}\hat{\boldsymbol{X}}) \boldsymbol{I} - (\boldsymbol{P}\hat{\boldsymbol{X}}) \hat{\boldsymbol{x}}^\top \Big), \\
(\hat{\boldsymbol{x}}' \times \tilde{\boldsymbol{P}}')^\top ((\boldsymbol{P}'\hat{\boldsymbol{X}}) \times \boldsymbol{I}) &= \tilde{\boldsymbol{P}}'^\top \Big((\hat{\boldsymbol{x}}', \boldsymbol{P}'\hat{\boldsymbol{X}}) \boldsymbol{I} - (\boldsymbol{P}'\hat{\boldsymbol{X}}) \hat{\boldsymbol{x}}'^\top \Big).
\end{aligned} \tag{44}
$$

Using these, we can rewrite Eq. (41) in the following form:

$$
\boldsymbol{A} \Delta\hat{\boldsymbol{r}} = \boldsymbol{B} \begin{pmatrix} \Delta\hat{\boldsymbol{x}} \\ \Delta\hat{\boldsymbol{x}}' \end{pmatrix},
$$

$$
\begin{aligned}
\boldsymbol{A} &\equiv \|\hat{\boldsymbol{x}}\|^2 \tilde{\boldsymbol{P}}^\top \boldsymbol{P}_{\mathcal{N}[\hat{\boldsymbol{x}}]} \tilde{\boldsymbol{P}} + \|\hat{\boldsymbol{x}}'\|^2 \tilde{\boldsymbol{P}}'^\top \boldsymbol{P}_{\mathcal{N}[\hat{\boldsymbol{x}}']} \tilde{\boldsymbol{P}}', \\
\boldsymbol{B} &\equiv \Big(\tilde{\boldsymbol{P}}^\top \Big((\hat{\boldsymbol{x}}, \boldsymbol{P}\hat{\boldsymbol{X}}) \boldsymbol{I} - (\boldsymbol{P}\hat{\boldsymbol{X}}) \hat{\boldsymbol{x}}^\top \\
&\qquad \tilde{\boldsymbol{P}}'^\top \Big((\hat{\boldsymbol{x}}', \boldsymbol{P}'\hat{\boldsymbol{X}}) \boldsymbol{I} - (\boldsymbol{P}'\hat{\boldsymbol{X}}) \hat{\boldsymbol{x}}'^\top \Big) \Big).
\end{aligned} \tag{45}
$$

Hence, we obtain

$$
\Delta\hat{\boldsymbol{r}} \Delta\hat{\boldsymbol{r}}^\top = \boldsymbol{A}^{-1} \boldsymbol{B} \begin{pmatrix} \Delta\hat{\boldsymbol{x}} \Delta\hat{\boldsymbol{x}}^\top & \Delta\hat{\boldsymbol{x}} \Delta\hat{\boldsymbol{x}}^\top \\ \Delta\hat{\boldsymbol{x}}' \Delta\hat{\boldsymbol{x}}^\top & \Delta\hat{\boldsymbol{x}}' \Delta\hat{\boldsymbol{x}}'^\top \end{pmatrix} \boldsymbol{B}^\top (\boldsymbol{A}^{-1})^\top. \tag{46}
$$

Taking expectation on both sides, we obtain the normalized covariance matrix $V_0[\hat{\boldsymbol{r}}]$ of the reconstructed position $\hat{\boldsymbol{r}}$ in the following form:

$$
V_0[\hat{\boldsymbol{r}}] = \boldsymbol{A}^{-1} \boldsymbol{B} \begin{pmatrix} V_0[\hat{\boldsymbol{x}}] & V_0[\hat{\boldsymbol{x}}, \hat{\boldsymbol{x}}'] \\ V_0[\hat{\boldsymbol{x}}', \hat{\boldsymbol{x}}] & V_0[\hat{\boldsymbol{x}}'] \end{pmatrix} \boldsymbol{B}^\top (\boldsymbol{A}^{-1})^\top. \tag{47}
$$

7. EXPERIMENTS

Our simulation setting is as follows (Fig. 2). A curved grid surface is rotated by angle 10° around an axis passing through the world coordinate origin O, and the 3-D position of each grid point is measured before and after the rotation by stereo vision. The grid is placed with its center at the origin O, and the two cameras are placed so that their lines of sight meet at O with angle 10°. Figure. 3 shows simulated images of the grid surface before and after the rotation. The image size is 500×800 pixels, and the focal length is set to 600 pixels. Gaussian noise of mean 0 and standard deviation σ pixels is independently added to the x and y coordinates of the grid points in the images, and their 3-D positions before and after the rotation are reconstructed by the method of Kanatani et al. [14] (Appendices B and C).

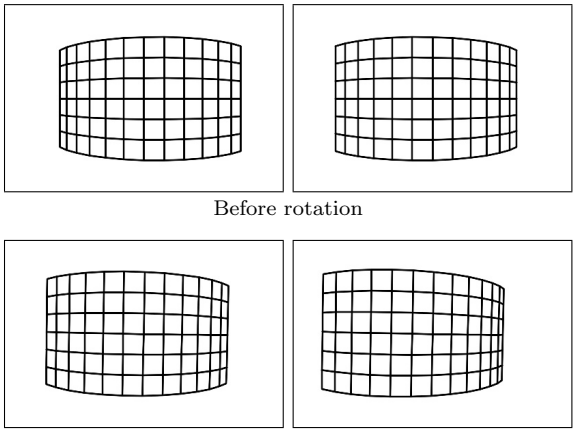

Figure 3: Simulated stereo images of the grid before and after the rotation.

Evaluating the normalized covariance matrix $V_0[\hat{\boldsymbol{r}}_\alpha]$ in Eq. (47), we find that the uncertainty distribution has an ellipsoidal shape elongated in the depth direction, as illustrated in Fig 2. The ratio of radii is, on average over all the points, 1.00 : 1.685 : 5.090 in the vertical, horizontal, and depth directions, respectively, meaning that the error in the depth direction is approximately five times as large as in the vertical direction. We actually measured this ratio by adding noise to the images many times and found that it is about 1.00 : 1.686 : 5.095, a very close value to the prediction by Eq. (47).

Using the thus predicted normalized covariance matrices $V_0[\hat{\boldsymbol{r}}_\alpha]$ and $V_0[\hat{\boldsymbol{r}}'_\alpha]$ before and after the rotation, we estimated the rotation of the grid surface and evaluated the deviation the quaternion $\hat{\boldsymbol{q}}$ of the computed rotation from its true value $\bar{\boldsymbol{q}}$ by

$$
\Delta\boldsymbol{q} = \boldsymbol{P}_{\bar{\boldsymbol{q}}} \hat{\boldsymbol{q}}, \qquad \boldsymbol{P}_{\bar{\boldsymbol{q}}} \equiv \boldsymbol{I} - \bar{\boldsymbol{q}} \bar{\boldsymbol{q}}^\top. \tag{48}
$$

Since $\hat{\boldsymbol{q}}$ is a unit vector, it is on a 3-D sphere S^3 in 4-D near $\bar{\boldsymbol{q}}$. We are interested only in the error component $\Delta\boldsymbol{q}$ of $\hat{\boldsymbol{q}}$ orthogonal to $\bar{\boldsymbol{q}}$, because there is no deviation in the direction of $\bar{\boldsymbol{q}}$ (Fig. 4). The matrix $\boldsymbol{P}_{\bar{\boldsymbol{q}}}$ in Eq. (48) orthogonally projects $\hat{\boldsymbol{q}}$ onto the tangent plane to S^3 at $\bar{\boldsymbol{q}}$. After 1000 independent trials using different nose each time, we evaluated the

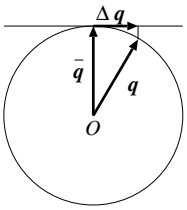

Figure 4: The component $\Delta\boldsymbol{q}$ of the computed quaternion vector $\hat{\boldsymbol{q}}$ orthogonal to its true value $\bar{\boldsymbol{q}}$.

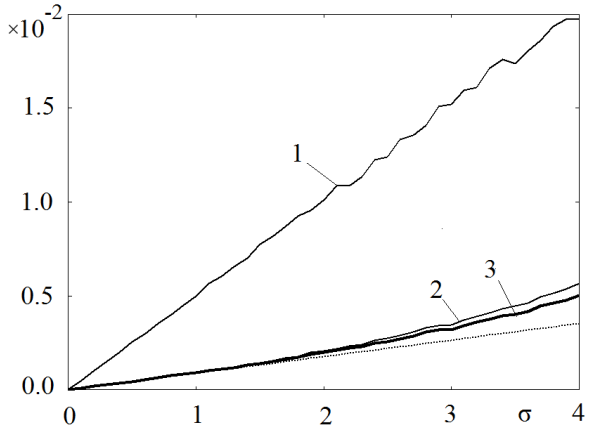

Figure 5: The RMS error of the computed rotation vs. the standard deviation σ of the noise added to the stereo images. The dotted line shows the KCR lower bound. 1. Optimal estimation assuming homogeneous isotropic noise. 2. Renormalization. 3. Proposed method.

root-mean-square (RMS) error

$$E = \sqrt{\frac{1}{1000}\sum_{a=1}^{1000} \|\Delta\boldsymbol{q}^{(a)}\|^2}, \tag{49}$$

where $\Delta\boldsymbol{q}^{(a)}$ is the ath value. The theoretical accuracy limit, called the *KCR lower bound* [12, 13], is given by

$$E_{\rm KCR} = \sigma {\rm tr}\Big(\sum_{\alpha=1}^{N} \bar{\boldsymbol{X}}_\alpha^\top \bar{\boldsymbol{W}}_\alpha \bar{\boldsymbol{X}}_\alpha\Big)^-, \tag{50}$$

where $\bar{\boldsymbol{X}}_\alpha$ and $\bar{\boldsymbol{W}}_\alpha$ are, respectively, the values of $\boldsymbol{X}_\alpha$ and $\boldsymbol{W}_\alpha$ when $\boldsymbol{q}$, $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$ in their defining equations are replaced by their true values $\bar{\boldsymbol{q}}$, $\bar{\boldsymbol{r}}_\alpha$, and $\bar{\boldsymbol{r}}'_\alpha$, respectively. The operation $(\,\cdot\,)^-$ means the pseudoinverse, and tr denotes the matrix trace.

Figure 5 plots the RMS error E for the standard deviation σ of the noise added to the stereo images, and the dotted line shows the KCR lower bound. We compared three methods:

1. The optimal method for homogeneous isotropic noise (Appendix A).
2. The renormalization of Ohta and Kanatani [18] (Appendix B).
3. The proposed method.

We can immediately see that the well known method for homogeneous isotropic noise performs very poorly. In contrast, the renormalization of Ohta and Kanatani [18] is confirmed to be highly accurate, nearly reaching the KCR lower bound. Yet, our proposed method is even better than renormalization, although the difference is very small.

8. CONCLUSIONS

We have presented a new method for optimally computing the 3-D rotation from two sets of 3-D data. Unlike 2-D data, the noise in 3-D data is inherently inhomogeneous and anisotropic, reflecting the 3-D sensing procedure. Following Ohta and Kanatani [18], we represented a 3-D rotation in terms of quaternion and derived a numerical procedure for computing an exact ML solution using the FNS of Chojnacki et al. [2]. We analyzed the uncertainty of 3-D reconstruction by stereo vision and optimally computed the 3-D rotation. It was shown that the widely used method, which assumes homogeneous and isotropic noise, is not suitable for 3-D data. We confirmed that the renormalization of Ohta and Kanatani [18] indeed computes almost an optimal solution and that, although the difference is small, the proposed method can compute an even better solution.

Acknowledgments. The authors thank Orhan Akyilmaz of Istanbul Institute of Technology, Turkey, and Naoya Ohta of Gunma University, Japan, for helpful discussions. They also thank our student Hiroki Hara for helping our numerical experiments. This work was supported in part by the Ministry of Education, Culture, Sports, Science, and Technology, Japan, under a Grant in Aid for Scientific Research (C 21500172).

References

[1] K. S. Arun, T. S. Huang and S. D. Blostein, Least squares fitting of two 3-D point sets, *IEEE Trans. Patt. Anal. Mach. Intell.*, **9**-5 (1987-5), 698–700.

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

[3] L. Dorst, First order error propagation of the Procrustes method for 3D attitude estimation, *IEEE Trans. Patt. Anal. Mach. Intell.*, **27**-2 (2005-2), 221–229.

[4] Y. A. Felus and R. C. Burch, On symmetrical three-dimensional datum conversion, *GPS Solutions* **13**-1 (2009-1), 65–74.

[5] D. Goryn and S. Hein, On the estimation of rigid body rotation from noisy data, *IEEE Trans. Patt. Anal. Mach. Intell.*, **17**-12 (1995-12), 1219–1200.

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

[7] B. K. P. Horn, Closed-form solution of absolute orientation, using quaternions, *Int. J. Opt. Soc. Am.* **A-4**-4 (1987-4), 629–642.

[8] B. K. P. Horn, H. M. Hildren and S. Negahdaripour, Closed-form solution of absolute orientation, using orthonormal matrices, *Int. J. Opt. Soc. Am.* **A-5**-7 (1988-7), 1127–1135.

[9] K. Kanatani, *Group-Theoretical Methods in Image Understanding*, Springer, Berlin, Germany, 1990.

[10] K. Kanatani, *Geometric Computation for Machine Vision*, Oxford University Press, Oxford, U.K., 1993.

[11] K. Kanatani, Analysis of 3-D rotation fitting, *IEEE Trans. Patt. Anal. Mach. Intell.*, **16**-5 (1994-5), 543–449.

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

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

[14] K. Kanatani, Y. Sugaya, and H. Niitsuma, Triangulation from two views revisited: Hartley-Sturm vs. optimal correction, *Proc. 19th British Machine Vision Conf.*, September 2008, Leeds, U.K., pp. 173–182.

[15] Y. Kanazawa and K. Kanatani, Reliability of 3-D reconstruction by stereo vision, *IEICE Trans. Inf. & Syst.*, **E78-D**-10 (1995-10), 1301–1306.

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

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

[18] N. Ohta and K. Kanatani, Optimal estimation of three-dimensional rotation and reliability evaluation, *IEICE Trans. Inf. & Syst.*, *E81-D*-11 (1998-11), 1247–1252.

[19] S. Umeyama, Least-squares estimation of transformation parameters between two point sets, *IEEE Trans. Patt. Anal. Mach. Intell.*, **13**-4 (1991-4), 379–380.

APPENDIX

A. Homogeneous Isotropic Noise Case

Various methods are known for optimally computing the 3-D rotation for homogeneous and isotropic noise [1, 7, 8, 11, 19], but all are mathematically equivalent. The simplest is the following method in terms of the singular value decomposition (SVD) [10]:

1. Compute the following correlation matrix $\boldsymbol{N}$ between the 3-D positions $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$ before and after the rotations:
$$\boldsymbol{N} = \sum_{\alpha=1}^{N} \boldsymbol{r}'_\alpha \boldsymbol{r}_\alpha^\top. \tag{51}$$
2. Compute the SVD of $\boldsymbol{N}$ in the form
$$\boldsymbol{N} = \boldsymbol{U}\mathrm{diag}(\sigma_1, \sigma_2, \sigma_3)\boldsymbol{V}^\top, \tag{52}$$
where $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal matrices, and $\sigma_1 \geq \sigma_2 \geq \sigma_3$ (≥ 0) are the singular values.
3. Return the following rotation matrix:
$$\boldsymbol{R} = \boldsymbol{U}\mathrm{diag}(1, 1, \det(\boldsymbol{U}\boldsymbol{V}^\top))\boldsymbol{V}^\top. \tag{53}$$

B. Renormalization

The renormalization procedure of Ohta and Kanatani [18] goes as follows:

1. Compute the matrix $\boldsymbol{X}_\alpha$ in Eq. (16) from the 3-D positions $\boldsymbol{r}_\alpha$ and $\boldsymbol{r}'_\alpha$ before and after the rotations.
2. Let $c = 0$ and $\boldsymbol{W}_\alpha = \boldsymbol{I}$ (3×3 unit matrix).
3. Compute the 4×4 matrix $\boldsymbol{M}$ in Eq. (19).
4. Compute the following scalar n, vector $\boldsymbol{n}$, and 3×3 matrix $\boldsymbol{N}'$:
$$\begin{aligned} n &= \sum_{\alpha=1}^{N} (\boldsymbol{W}_\alpha; V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]), \\ \boldsymbol{n} &= 2\sum_{\alpha=1}^{N} \mathrm{vec}[\mathcal{A}[\boldsymbol{W}_\alpha(V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])]], \\ \boldsymbol{N}' &= \sum_{\alpha=1}^{N} [\boldsymbol{W}_\alpha \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha])]. \end{aligned} \tag{54}$$

Here, the inner product $(\boldsymbol{A}; \boldsymbol{B})$ of matrices $\boldsymbol{A} = (A_{ij})$ and $\boldsymbol{B} = (B_{ij})$ is defined by $(\boldsymbol{A}; \boldsymbol{B}) = \sum_{i,j=1}^{3} A_{ij}B_{ij}$, and the operator $\mathcal{A}[\,\cdot\,]$ means antisymmetrization ($\mathcal{A}[\boldsymbol{A}] = (\boldsymbol{A} - \boldsymbol{A}^\top)/2$). For an antisymmetric matrix $\boldsymbol{A} = (A_{ij})$ ($\boldsymbol{A}^\top = -\boldsymbol{A}$), we define the vectorization operation $\mathrm{vec}[\,\cdot\,]$ by $\mathrm{vec}[\,\cdot\,] = (A_{32}, A_{13}, A_{21})^\top$. We also define the

exterior product $[\boldsymbol{A} \times \boldsymbol{B}]$ of matrices $\boldsymbol{A} = (A_{ij})$ and $\boldsymbol{B} = (B_{ij})$ to be a matrix whose (ij) element is $\sum_{k,l,m,n=1}^{3} \varepsilon_{ikl}\varepsilon_{jmn}A_{km}B_{ln}$, where ε_{ijk} is a permutation symbol, assuming 1 when (ijk) is an even permutation of (123), -1 when it is an odd permutation, and 0 otherwise (i.e., with repeated indices).

5. Compute the following 4×4 matrix $\boldsymbol{N}$:

$$\boldsymbol{N} = \begin{pmatrix} n & \boldsymbol{n}^\top \\ \boldsymbol{n} & \boldsymbol{N}' \end{pmatrix}. \tag{55}$$

6. Solve the eigenvalue problem

$$(\boldsymbol{M} - c\boldsymbol{N})\boldsymbol{q} = \lambda \boldsymbol{q}, \tag{56}$$

and compute the unit eigenvector $\boldsymbol{q}$ corresponding to the smallest eigenvalue λ.

7. If $|\lambda| \approx 0$, return $\boldsymbol{q}$ and stop. Else, go back to Step 3 after updating c and $\boldsymbol{W}_\alpha$ by

$$\begin{aligned} c &\leftarrow c + \frac{\lambda}{(\boldsymbol{q}, \boldsymbol{N}\boldsymbol{q})}, \\ \boldsymbol{W}_\alpha &\leftarrow \Big(q_0^2(V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \\ &\quad -2q_0\mathcal{S}[\boldsymbol{q_l} \times (V_0[\boldsymbol{r}'_\alpha] - V_0[\boldsymbol{r}_\alpha])] \\ &\quad +\boldsymbol{q_l} \times (V_0[\boldsymbol{r}'_\alpha] + V_0[\boldsymbol{r}_\alpha]) \times \boldsymbol{q_l}\Big)^{-1}. \end{aligned} \tag{57}$$

C. 3-D Reconstruction by Stereo Vision

If a point (x, y) in the first image of a stereo pair corresponds to (x', y') in the second, they satisfy the following *epipolar constraint* [6]:

$$\left(\begin{pmatrix} x/f_0 \\ y/f_0 \\ 1 \end{pmatrix}, \boldsymbol{F} \begin{pmatrix} x'/f_0 \\ y'/f_0 \\ 1 \end{pmatrix}\right) = 0, \tag{58}$$

Here, the matrix $\boldsymbol{F} = (F_{ij})$, called the *fundamental matrix*, is defined by

$$\begin{aligned} F_{11} &= \begin{vmatrix} P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P'_{21} & P'_{22} & P'_{23} & P'_{24} \\ P'_{31} & P'_{32} & P'_{33} & P'_{34} \end{vmatrix}, \quad F_{12} = \begin{vmatrix} P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P'_{31} & P'_{32} & P'_{33} & P'_{34} \\ P'_{11} & P'_{12} & P'_{13} & P'_{14} \end{vmatrix}, \\ F_{13} &= \begin{vmatrix} P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \\ P'_{11} & P'_{12} & P'_{13} & P'_{14} \\ P'_{21} & P'_{22} & P'_{23} & P'_{24} \end{vmatrix}, \quad F_{21} = \begin{vmatrix} P_{31} & P_{32} & P_{33} & P_{34} \\ P_{11} & P_{12} & P_{13} & P_{14} \\ P'_{21} & P'_{22} & P'_{23} & P'_{24} \\ P'_{31} & P'_{32} & P'_{33} & P'_{34} \end{vmatrix}, \\ F_{22} &= \begin{vmatrix} P_{31} & P_{32} & P_{33} & P_{34} \\ P_{11} & P_{12} & P_{13} & P_{14} \\ P'_{31} & P'_{32} & P'_{33} & P'_{34} \\ P'_{11} & P'_{12} & P'_{13} & P'_{14} \end{vmatrix}, \quad F_{23} = \begin{vmatrix} P_{31} & P_{32} & P_{33} & P_{34} \\ P_{11} & P_{12} & P_{13} & P_{14} \\ P'_{11} & P'_{12} & P'_{13} & P'_{14} \\ P'_{21} & P'_{22} & P'_{23} & P'_{24} \end{vmatrix}, \\ F_{31} &= \begin{vmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P'_{21} & P'_{22} & P'_{23} & P'_{24} \\ P'_{31} & P'_{32} & P'_{33} & P'_{34} \end{vmatrix}, \quad F_{32} = \begin{vmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P'_{31} & P'_{32} & P'_{33} & P'_{34} \\ P'_{11} & P'_{12} & P'_{13} & P'_{14} \end{vmatrix}, \\ F_{33} &= \begin{vmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P'_{11} & P'_{12} & P'_{13} & P'_{14} \\ P'_{21} & P'_{22} & P'_{23} & P'_{24} \end{vmatrix}, \end{aligned} \tag{59}$$

where P_{ij} and P'_{ij} are the (ij) elements of the projection matrices $\boldsymbol{P}$ and $\boldsymbol{P}'$ of the first and the second camera, respectively, as defined in Eq. (36). If we let $\boldsymbol{r} = (X, Y, Z)^\top$ be the 3-D point we are looking at, we obtain from the perspective projection relationship in Eqs. (35)

$$\begin{aligned} x &= f_0 \frac{P_{11}X + P_{12}X + P_{13}X + P_{14}f_0}{P_{31}X + P_{32}X + P_{33}X + P_{34}f_0}, \\ y &= f_0 \frac{P_{21}X + P_{22}X + P_{23}X + P_{24}f_0}{P_{31}X + P_{32}X + P_{33}X + P_{34}f_0}, \\ x' &= f_0 \frac{P'_{11}X + P'_{12}X + P'_{13}X + P'_{14}f_0}{P'_{31}X + P'_{32}X + P'_{33}X + P'_{34}f_0}, \\ y' &= f_0 \frac{P'_{21}X + P'_{22}X + P'_{23}X + P'_{24}f_0}{P'_{31}X + P'_{32}X + P'_{33}X + P'_{34}f_0}. \end{aligned} \tag{60}$$

Clearing the fraction, we obtain the following simultaneous linear equations:

$$\begin{aligned} &\begin{pmatrix} xP_{31} - f_0P_{11} & xP_{32} - f_0P_{12} & xP_{33} - f_0P_{13} \\ yP_{31} - f_0P_{21} & yP_{32} - f_0P_{22} & yP_{33} - f_0P_{23} \\ x'P'_{31} - f_0P'_{11} & x'P'_{32} - f_0P'_{12} & x'P'_{33} - f_0P'_{13} \\ y'P'_{31} - f_0P'_{21} & y'P'_{32} - f_0P'_{22} & y'P'_{33} - f_0P'_{23} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \\ &= -\begin{pmatrix} xP_{34} - f_0P_{14} \\ yP_{34} - f_0P_{24} \\ x'P'_{34} - f_0P'_{14} \\ y'P'_{34} - f_0P'_{24} \end{pmatrix}. \end{aligned} \tag{61}$$

These are four equations for three unknowns X, Y, and Z, but because the epipolar constraint in Eq. (58) is satisfied, the solution is unique. In fact, Eq. (58) is derived as the necessary and sufficient condition for Eq. (61) to have a unique solution.

D. Optimal Triangulation

Let (x, y) and (x', y') be a pair of corresponding points between stereo images. Since correspondence detection by an image processing operations inevitably entails uncertainty to some degree, they do not necessarily satisfy the epipolar constraint in Eq. (58). Geometrically, this corresponds to the fact that the lines of sight starting from the lens center of the two cameras and passing through (x, y) and (x', y') in the image plane do not necessarily meet in the scene (Fig. 6). For optimal 3-D reconstruction, we need to correct (x, y) and (x', y') optimally to $(\hat{x}, \hat{y})$ and $(\hat{x}', \hat{y}')$ so that their lines of sight intersect, i.e., Eq. (58) is satisfied. By "optimally", we mean that the correction is done in such a way that the *reprojection error* $(\hat{x} - x)^2 + (\hat{y} - y)^2 + (\hat{x}' - x')^2 + (\hat{y}' - y')^2$ is minimized. This correction procedure goes as follows [14]:

1. Let $E_0 = \infty$ (a sufficiently large number), $\hat{x} = x$, $\hat{y} = y$, $\hat{x}' = x'$, $\hat{y}' = y'$, and $\tilde{x} = \tilde{y} = \tilde{x}' = \tilde{y}' = 0$, and express the fundamental matrix $\boldsymbol{F} =$

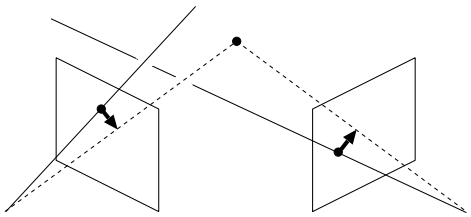

Figure 6: The detected corresponding points are optimally corrected so that their lines of sight intersect.

(F_{ij}) in the 9-D vector form

$$\boldsymbol{f} = (F_{11}, F_{12}, F_{13}, F_{21}, F_{22}, F_{23}, F_{31}, F_{32}, F_{33})^\top. \tag{62}$$

2. Compute the following 9×9 matrix $V_0[\hat{\boldsymbol{\xi}}]$ and the 9-D vector $\boldsymbol{\xi}^*$:

$$V_0[\hat{\boldsymbol{\xi}}] = \begin{pmatrix} \hat{x}^2+\hat{x}'^2 & \hat{x}'\hat{y}' & f_0\hat{x}' & \hat{x}\hat{y} \\ \hat{x}'\hat{y}' & \hat{x}^2+\hat{y}'^2 & f_0\hat{y}' & 0 \\ f_0\hat{x}' & f_0\hat{y}' & f_0^2 & 0 \\ \hat{x}\hat{y} & 0 & 0 & \hat{y}^2+\hat{x}'^2 \\ 0 & \hat{x}\hat{y} & 0 & \hat{x}'\hat{y}' \\ 0 & 0 & 0 & f_0\hat{x}' \\ f_0\hat{x} & 0 & 0 & f_0\hat{y} \\ 0 & f_0\hat{x} & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & f_0\hat{x} & 0 & 0 \\ \hat{x}\hat{y} & 0 & 0 & f_0\hat{x} & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \hat{x}'\hat{y}' & f_0\hat{x}' & f_0\hat{y} & 0 & 0 \\ \hat{y}^2+\hat{y}'^2 & f_0\hat{y}' & 0 & f_0\hat{y} & 0 \\ f_0\hat{y}' & f_0^2 & 0 & 0 & 0 \\ 0 & 0 & f_0^2 & 0 & 0 \\ f_0\hat{y} & 0 & 0 & f_0^2 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{63}$$

$$\boldsymbol{\xi}^* = \begin{pmatrix} \hat{x}\hat{x}'+\hat{x}'\tilde{x}+\hat{x}\tilde{x}' \\ \hat{x}\hat{y}'+\hat{y}'\tilde{x}+\hat{x}\tilde{y}' \\ \hat{x}+\tilde{x} \\ \hat{y}\hat{x}'+\hat{x}'\tilde{y}+\hat{y}\tilde{x}' \\ \hat{y}\hat{y}'+\hat{y}'\tilde{y}+\hat{y}\tilde{y}' \\ \hat{y}+\tilde{y} \\ \hat{x}'+\tilde{x}' \\ \hat{y}'+\tilde{y}' \\ 1 \end{pmatrix}. \tag{64}$$

3. Update $\tilde{x}$, $\tilde{y}$, $\tilde{x}'$, $\tilde{y}'$, $\hat{x}$, $\hat{y}$, $\hat{x}'$, and $\hat{y}'$ as follows:

$$\begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} \leftarrow \frac{(\boldsymbol{f}, \boldsymbol{\xi}^*)}{(\boldsymbol{f}, V_0[\hat{\boldsymbol{\xi}}]\boldsymbol{f})} \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \end{pmatrix} \begin{pmatrix} \hat{x}' \\ \hat{y}' \\ 1 \end{pmatrix},$$

$$\begin{pmatrix} \tilde{x}' \\ \tilde{y}' \end{pmatrix} \leftarrow \frac{(\boldsymbol{f}, \boldsymbol{\xi}^*)}{(\boldsymbol{f}, V_0[\hat{\boldsymbol{\xi}}]\boldsymbol{f})} \begin{pmatrix} F_{11} & F_{21} & F_{31} \\ F_{12} & F_{22} & F_{32} \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{y} \\ 1 \end{pmatrix}, \tag{65}$$

$$\hat{x} \leftarrow x - \tilde{x}, \qquad \hat{y} \leftarrow y - \tilde{y},$$
$$\hat{x}' \leftarrow x' - \tilde{x}', \qquad \hat{y}' \leftarrow y' - \tilde{y}'. \tag{66}$$

4. Compute the reprojection error E by

$$E = \tilde{x}^2 + \tilde{y}^2 + \tilde{x}'^2 + \tilde{y}'^2. \tag{67}$$

If $E \approx E_0$, return $(\hat{x}, \hat{y})$ and $(\hat{x}', \hat{y}')$ and stop. Else, let $E_0 \leftarrow E$ and go back to Step 2.