PAPER Special Issue on Machine Vision Applications

Optimal Structure-from-Motion Algorithm for Optical Flow

Naoya OHTA† *and* Kenichi KANATANI†, *Members*

SUMMARY This paper presents a new method for solving the structure-from-motion problem for optical flow. The fact that the structure-from-motion problem can be simplified by using the linearization technique is well known. However, it has been pointed out that the linearization technique reduces the accuracy of the computation. In this paper, we overcome this disadvantage by correcting the linearized solution in a statistically optimal way. Computer simulation experiments show that our method yields an unbiased estimator of the motion parameters which almost attains the theoretical bound on accuracy. Our method also enables us to evaluate the reliability of the reconstructed structure in the form of the covariance matrix. Real-image experiments are conducted to demonstrate the effectiveness of our method.

key words: *optical flow, motion parameter, depth map, maximum likelihood estimation, renormalization*

1. Introduction

The *structure-from-motion* problem for computing 3-D from optical flow has been intensively investigated by many researchers ever since Ullman [11] addressed this theme. If the object has a parameterized surface, such as a polynomial or planar patch, we only need to determine the coefficients, and the solution can be analytically obtained. For a surface of general shape, however, it is difficult to obtain an analytical solution. Zhuang et al. [13] applied the *linearization technique*, which was first developed for the finite motion analysis [12], to optical flow. However, this technique sacrifices the accuracy for the computational ease. Some researchers formulated the problem in the form of the least-squares minimization and solved it by a numerical search algorithm. However, it has been pointed out that the resulting solution has a systematic bias [2]. An unbiased solution can be obtained by computing the maximum likelihood solution, but the computational burden is heavy [1], [3], [9]. In order to avoid this difficulty, several researchers have devised simplified methods for obtaining an unbiased estimator [5], [10]. In this report, we modify our previously proposed method which used the *renormalization scheme* [5]. Our new method uses the linarization technique by regarding the *flow matrix* as a new variable and corrects the linearized solution in a statistically optimal way. The estimator of our method is shown to be not only unbiased but also very accurate to the extent that the theoretical bound is almost attained. Original notations and definitions about vectors, matrices and tensors are listed in Appendix [7].

Manuscript received July 20, 1995.

†The authors are with the Department of Computer Science, Gunma University, Kiryu-shi, 376 Japan.

2. The Problem and Definitions

We fix an XYZ Cartesian coordinate system to the camera. The optical axis of the camera coincides with the Z-axis, and the focal length is set to 1 without losing generality, so the image plane is $Z = 1$. The camera is assumed to be moving in a stationary environment with translation velocity $\boldsymbol{v}$ and rotation velocity $\boldsymbol{\omega}$; we call $\{\boldsymbol{v}, \boldsymbol{\omega}\}$ the *motion parameters*. The points in the image at which optical flow is observed are indexed by subscript $\alpha = 1, 2, \ldots, n$. The image coordinates (x_α, y_α) of the αth point are represented by a three-dimensional vector $\boldsymbol{x}_\alpha = (x_\alpha \ y_\alpha \ 1)^\top$, and the optical flow $(\dot{x}_\alpha, \dot{y}_\alpha)$ defined there is represented by $\dot{\boldsymbol{x}}_\alpha = (\dot{x}_\alpha \ \dot{y}_\alpha \ 0)^\top$.

Let $\bar{\dot{\boldsymbol{x}}}_\alpha$ be the optical flow that should be observed in the absence of noise. It is easy to show that the following *epipolar constraint* is satisfied [5]:

$$|\boldsymbol{x}_\alpha, \bar{\dot{\boldsymbol{x}}}_\alpha + \boldsymbol{v} + \boldsymbol{\omega} \times \boldsymbol{x}_\alpha, \boldsymbol{v}| = 0. \tag{1}$$

Here, $|\cdot, \cdot, \cdot|$ denotes the scalar triple product of vectors. Since the absolute magnitude of the translation velocity $\boldsymbol{v}$ is indeterminate, we adopt the normalization

$$\|\boldsymbol{v}\| = 1, \tag{2}$$

assuming that $\boldsymbol{v} \neq \boldsymbol{0}$ (it is easy to test if $\boldsymbol{v} \neq \boldsymbol{0}$; if $\boldsymbol{v} = \boldsymbol{0}$, no 3-D information is available). In the presence of noise, an observed flow $\dot{\boldsymbol{x}}_\alpha$ does not necessarily satisfy Eq. (1). We model the noise as an additive Gaussian random variable $\Delta\dot{\boldsymbol{x}}_\alpha = (\Delta\dot{x}_\alpha \ \Delta\dot{y}_\alpha \ 0)^\top$ and write

$$\dot{\boldsymbol{x}}_\alpha = \bar{\dot{\boldsymbol{x}}}_\alpha + \Delta\dot{\boldsymbol{x}}_\alpha. \tag{3}$$

The mean and the covariance matrix of the noise $\Delta\dot{\boldsymbol{x}}_\alpha$ are assumed to be $\boldsymbol{0}$ and $\boldsymbol{V}_\alpha$, respectively. Since the third component of $\dot{\boldsymbol{x}}_\alpha$ is 0, the covariance matrix $\boldsymbol{V}_\alpha$ is singular; the third low and the third column consist of 0. The noise $\Delta\dot{\boldsymbol{x}}_\alpha$ is assumed to be independent for different α.

Given an optical flow $\dot{\boldsymbol{x}}_\alpha$, $\alpha = 1, 2, \ldots, n$, the parameters to be estimated are $\boldsymbol{v}$, $\boldsymbol{\omega}$ and $\bar{\dot{\boldsymbol{x}}}_\alpha$, $\alpha = 1, 2, \ldots, n$, which are constrained by Eqs. (1) and (2). In this paper, we adopt the *maximum likelihood criterion*: we minimize the log-likelihood

$$J_1 = \sum_{\alpha=1}^{n} (\dot{\boldsymbol{x}}_\alpha - \bar{\dot{\boldsymbol{x}}}_\alpha, (\boldsymbol{V}_\alpha)_2^- (\dot{\boldsymbol{x}}_\alpha - \bar{\dot{\boldsymbol{x}}}_\alpha)), \tag{4}$$

where $(\cdot, \cdot)$ denotes the inner product of vectors and $(\cdot)_2^-$ denotes the *rank-constrained generalized inverse* (see Appendix). The value $\bar{\dot{\boldsymbol{x}}}_\alpha$ that minimizes J_1 under the constraints (1) and (2) is easily obtained (we will discuss this in Sect. 6). Substituting the value $\bar{\dot{\boldsymbol{x}}}_\alpha$ thus computed into Eq. (4), we obtain another objective function

$$J_2 = \sum_{\alpha=1}^{n} \frac{|\boldsymbol{x}_\alpha, \dot{\boldsymbol{x}}_\alpha + \boldsymbol{v} + \boldsymbol{\omega} \times \boldsymbol{x}_\alpha, \boldsymbol{v}|^2}{(\boldsymbol{v} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha (\boldsymbol{v} \times \boldsymbol{x}_\alpha))}. \tag{5}$$

Minimization of J_2 is constrained by Eq. (2) alone.

Function J_1 or J_2 can be minimized by a numerical algorithm [3], [9]. However, computation is often unstable because J_1 and J_2 are complicated nonlinear functions. In order to avoid this difficulty, we apply the *linearization technique*. Define the *observation matrix* $\boldsymbol{X}_\alpha$ and the *flow matrix* $\boldsymbol{F}$ by

$$\boldsymbol{X}_\alpha = \boldsymbol{x}_\alpha \boldsymbol{x}_\alpha^\top + A[\dot{\boldsymbol{x}}_\alpha \boldsymbol{x}_\alpha^\top], \tag{6}$$

$$\boldsymbol{F} = (\boldsymbol{v}, \boldsymbol{\omega}) \boldsymbol{I} + S[\boldsymbol{v}\boldsymbol{\omega}^\top] + \boldsymbol{v} \times \boldsymbol{I}, \tag{7}$$

where $\boldsymbol{I}$ denotes the unit matrix and $S[\cdot]$ and $A[\cdot]$ are the symmetrization and anti-symmetrization operators, respectively (see Appendix). The observation matrix $\boldsymbol{X}_\alpha$ is determined by the observed optical flow alone, while the flow matrix $\boldsymbol{F}$ is determined by the motion parameters alone. Using these definitions, we can rewrite the epipolar constraint (1) in the form

$$(\bar{\boldsymbol{X}}_\alpha; \boldsymbol{F}) = 0, \tag{8}$$

where $(\cdot\,; \cdot)$ denotes the matrix inner product (see Appendix). In the above equation, $\bar{\boldsymbol{X}}_\alpha$ denotes the true observation matrix obtained by replacing $\dot{\boldsymbol{x}}$ by $\bar{\dot{\boldsymbol{x}}}$ in Eq. (6). We now have a new problem: given $\boldsymbol{X}_\alpha$, estimate $\boldsymbol{F}$ and $\bar{\boldsymbol{X}}_\alpha$. Since Eq. (6) is linear in $\dot{\boldsymbol{x}}_\alpha$, the noise in $\boldsymbol{X}_\alpha$ is also a Gaussian random variable of mean $\boldsymbol{0}$. Let $\mathcal{V}_\alpha$ be its covariance tensor: its $(ijkl)$ element is the covariance between the (ij) and (kl) elements of $\boldsymbol{X}_\alpha$ and given by

$$\begin{aligned} \mathcal{V}_{\alpha ijkl} = \frac{1}{4}(&V_{\alpha ik} x_{\alpha j} x_{\alpha l} - V_{\alpha il} x_{\alpha j} x_{\alpha k} \\ &- V_{\alpha jk} x_{\alpha i} x_{\alpha l} + V_{\alpha jl} x_{\alpha i} x_{\alpha k}). \end{aligned} \tag{9}$$

Then, minimizing Eq. (5) is equivalent to minimizing

$$J_3 = \sum_{\alpha=1}^{n} \frac{(\boldsymbol{X}_\alpha; \boldsymbol{F})^2}{(\boldsymbol{F}; \mathcal{V}_\alpha \boldsymbol{F})}. \tag{10}$$

The constraint (2) is equivalently rewritten in the form

$$\|A[\boldsymbol{F}]\| = \sqrt{2}. \tag{11}$$

In addition, another constraint arises. The flow matrix $\boldsymbol{F}$ is determined by $\boldsymbol{v}$ and $\boldsymbol{\omega}$, which have three elements each. Since $\boldsymbol{F}$ has nine elements, there remain three extra degrees of freedom, which originate from the fact that not all matrices can be decomposed into $\boldsymbol{v}$ and $\boldsymbol{\omega}$ in the form of Eq. (7). The condition for $\boldsymbol{F}$ to have the form of Eq. (7), which we call the *decomposability condition*, is expressed by the following equation [5]:

$$\boldsymbol{K} - \frac{1}{2}(\mathrm{tr}\boldsymbol{K})(\boldsymbol{I} - \boldsymbol{v}\boldsymbol{v}^\top) - 2S[\boldsymbol{K}\boldsymbol{v}\boldsymbol{v}^\top] = \boldsymbol{O}. \tag{12}$$

Here,

$$\boldsymbol{K} = S[\boldsymbol{F}], \quad \boldsymbol{v} = (A[\boldsymbol{F}]_{32}\ A[\boldsymbol{F}]_{13}\ A[\boldsymbol{F}]_{21})^\top, \tag{13}$$

where $A[\boldsymbol{F}]_{ij}$ denotes the (ij) element of $A[\boldsymbol{F}]$.

Proof: If Eq. (7) holds, Eq. (12) can be confirmed by substituting Eq. (7) into Eq. (13). Conversely, if Eq. (12) holds, we have $(\boldsymbol{v}, \boldsymbol{K}\boldsymbol{v}) = 0$. Equation (7) can be confirmed by substituting the following vector $\boldsymbol{\omega}$ into the right-hand side of Eq. (7):

$$\boldsymbol{\omega} = \frac{1}{2}(\mathrm{tr}\boldsymbol{K})\boldsymbol{v} - 2\boldsymbol{K}\boldsymbol{v}. \tag{14}$$

3. Minimization by Renormalization

The function J_3 should be minimized under the constraint (11) and the decomposability condition (12). In the following, we temporarily ignore the decomposability condition and compute the flow matrix $\boldsymbol{F}$ by a numerical scheme called *renormalization*. The computed flow matrix $\boldsymbol{F}$ is then corrected so as to satisfy Eq. (12) in a statistically optimal way.

If we use renormalization, we need not know the absolute magnitude of the covariance matrix $\boldsymbol{V}_\alpha$; we only need an appropriately scaled value $\boldsymbol{V}_\alpha^0$, which we call the *normalized covariance matrix*. The relative ratio of the two, which we denote by ϵ^2, is estimated a posteriori after the renormalization process. We call ϵ the *noise level*. The *normalized covariance tensor* $\mathcal{V}_\alpha$ is also defined in the same way:

$$\boldsymbol{V}_\alpha = \epsilon^2 \boldsymbol{V}_\alpha^0, \quad \mathcal{V}_\alpha = \epsilon^2 \mathcal{V}_\alpha^0. \tag{15}$$

The renormalization procedure is described as follows:

Step 1: Read optical flow $\dot{\boldsymbol{x}}_\alpha$ and the normalized covariance matrices $\boldsymbol{V}_\alpha^0$, $\alpha = 1, 2, \ldots, n$. Convert them into $\boldsymbol{X}_\alpha$ and $\mathcal{V}_\alpha^0$ by Eqs. (6) and (9). Set $W_\alpha = 1$ and $c = 0$.

Step 2: Compute tensors $\mathcal{M} = (\mathcal{M}_{ijkl})$ and $\mathcal{N} = (\mathcal{N}_{ijkl})$ by

$$\mathcal{M}_{ijkl} = \frac{1}{n} \sum_{\alpha=1}^{n} W_\alpha \boldsymbol{X}_{\alpha ij} \boldsymbol{X}_{\alpha kl}, \tag{16}$$

$$\mathcal{N} = \frac{1}{n} \sum_{\alpha=1}^{n} W_\alpha \mathcal{V}_\alpha^0. \tag{17}$$

Then, compute the *unbiased moment tensor*

$$\hat{\mathcal{M}} = \mathcal{M} - c\mathcal{N}. \tag{18}$$

Step 3: Compute the smallest eigenvalue λ_0 of tensor $\hat{\mathcal{M}}$ and its eigenmatrix $\hat{\boldsymbol{F}}$ (see Appendix). Then, normalize $\hat{\boldsymbol{F}}$ into $\|A[\hat{\boldsymbol{F}}]\| = \sqrt{2}$. If $\lambda_0 \approx 0$, return $\hat{\mathcal{M}}$, $\hat{\boldsymbol{F}}$ and c, and stop.

Step 4: Update W_α and c by the following equations, and go back to the Step 2:

$$c \leftarrow c + \frac{\lambda_0 \|\hat{\boldsymbol{F}}\|^2}{(\hat{\boldsymbol{F}}; \mathcal{N}\hat{\boldsymbol{F}})}, \quad W_\alpha \leftarrow \frac{1}{(\hat{\boldsymbol{F}}; \mathcal{V}_\alpha^0 \hat{\boldsymbol{F}})}. \tag{19}$$

The returned matrix $\hat{\boldsymbol{F}}$ is an optimal estimate of the flow matrix. The normalized covariance tensor of $\hat{\boldsymbol{F}}$, which we denote by $\mathcal{V}_F^0$, is computed as follows. Define a *projection tensor* $\mathcal{P} = (\mathcal{P}_{ijkl})$ by

$$\mathcal{P}_{ijkl} = \delta_{ik}\delta_{jl} - \frac{1}{2} A[\hat{\boldsymbol{F}}]_{ij} A[\hat{\boldsymbol{F}}]_{kl}, \tag{20}$$

where δ_{ij} is the *Kronecker delta*. Define a tensor $\hat{\mathcal{M}}' = (\hat{\mathcal{M}}'_{ijkl})$ by

$$\hat{\mathcal{M}}'_{ijkl} = \sum_{m,n,p,q=1}^{3} \mathcal{P}_{ijmn}\mathcal{P}_{klpq}\hat{\mathcal{M}}_{mnpq}. \tag{21}$$

This is a projection of the tensor $\hat{\mathcal{M}}$ onto the linear subspace compatible with the constraint (11). The normalized covariance tensor $\mathcal{V}_F^0$ is the following equation:

$$\mathcal{V}_F^0 = \frac{1}{n}(\hat{\mathcal{M}}')_5^-. \tag{22}$$

4. Correction of the Flow Matrix and Decomposition into the Motion Parameters

We have now obtained an estimate $\hat{\boldsymbol{F}}$ of the flow matrix and its normalized covariance tensor $\mathcal{V}_F^0$. What we need to do next is correct $\hat{\boldsymbol{F}}$ so as to satisfy the decomposability condition (12) in a statistically optimal way. Since the decomposability condition is nonlinear, it is difficult to compute an analytical solution. Here, we linearize the constraint in the neighborhood of $\hat{\boldsymbol{F}}$ and obtain an analytical solution. We iterate this process until the constraint holds sufficiently.

Let $\boldsymbol{D}$ be the left-hand side of Eq. (12). If we write $\boldsymbol{F} = \hat{\boldsymbol{F}} + \Delta\boldsymbol{F}$, the constraint (12) is linearized as follows:

$$\mathcal{C}\Delta\boldsymbol{F} = \boldsymbol{D}, \tag{23}$$

$$\mathcal{C}_{ijkl} = -\frac{1}{2}\sum_{m=1}^{3} \epsilon_{klm}\mathcal{A}_{ijm} + \mathcal{B}_{ijkl}, \tag{24}$$

$$\mathcal{A}_{ijk} = \frac{1}{2}(\mathrm{tr}\boldsymbol{K})(\delta_{ik}\boldsymbol{v}_j + \delta_{jk}\boldsymbol{v}_i) - \boldsymbol{K}_{ik}\boldsymbol{v}_j - \boldsymbol{K}_{jk}\boldsymbol{v}_i - \delta_{jk}\sum_{l=1}^{3}\boldsymbol{K}_{il}\boldsymbol{v}_l - \delta_{ik}\sum_{l=1}^{3}\boldsymbol{K}_{jl}\boldsymbol{v}_l, \tag{25}$$

$$\mathcal{B}_{ijkl} = \frac{1}{2}(\delta_{ik}\delta_{jl} + \delta_{il}\delta_{jk} - \delta_{ij}\delta_{kl} + \delta_{kl}\boldsymbol{v}_i\boldsymbol{v}_j - \delta_{ik}\boldsymbol{v}_l\boldsymbol{v}_j - \delta_{jl}\boldsymbol{v}_k\boldsymbol{v}_i - \delta_{jl}\boldsymbol{v}_k\boldsymbol{v}_i - \delta_{il}\boldsymbol{v}_k\boldsymbol{v}_j - \delta_{jk}\boldsymbol{v}_l\boldsymbol{v}_i). \tag{26}$$

Here, ϵ_{ijk} is the *Eddington epsilon*.

We compute the statistically most likely value of $\Delta\boldsymbol{F}$ by minimizing $(\Delta\boldsymbol{F}; (\mathcal{V}_F^0)_8^- \Delta\boldsymbol{F})$ under the constraint (23) together with Eq. (11), which is also linearized. The solution $\Delta\boldsymbol{F}$ has the following form:

$$\Delta\boldsymbol{F}_{ij} = \sum_{k,l,m,n,p,q=1}^{3} \mathcal{W}_{mnpq}\mathcal{C}_{mnkl}\mathcal{V}^0_{F\,ijkl}\boldsymbol{D}_{pq}, \tag{27}$$

$$\mathcal{W} = (\mathcal{V})_3^-, \tag{28}$$

$$\mathcal{V}_{ijkl} = \sum_{m,n,p,q=1}^{3} \mathcal{C}_{ijmn}\mathcal{C}_{klpq}\mathcal{V}^0_{F\,mnpq}. \tag{29}$$

However, $\hat{\boldsymbol{F}} - \Delta\boldsymbol{F}$ may not satisfy Eq. (11) exactly, since we have used a linear approximation. Here, we correct $\hat{\boldsymbol{F}}$ in the form

$$\hat{\boldsymbol{F}} \leftarrow \frac{\sqrt{2}(\hat{\boldsymbol{F}} - \Delta\boldsymbol{F})}{\|A[\hat{\boldsymbol{F}} - \Delta\boldsymbol{F}]\|}, \tag{30}$$

so that Eq. (11) is exactly satisfied. Furthermore, tensor $\mathcal{V}_F^0$ must also be corrected so that its null space is compatible with the constraint (11). This is done in the following form, where $\mathcal{P} = (\mathcal{P}_{ijkl})$ is the projection tensor defined by Eq. (20):

$$\mathcal{V}^0_{F\,ijkl} \leftarrow \sum_{m,n,p,q=1}^{3} \mathcal{P}_{ijmn}\mathcal{P}_{klpq}\mathcal{V}^0_{F\,mnpq}. \tag{31}$$

Figure 1 schematically shows the above process. The thick curve represents the constraint $\boldsymbol{D} = \boldsymbol{O}$, and the dotted curves represent the contours $\boldsymbol{D} =$ constant. The matrix $\hat{\boldsymbol{F}}$ is at point a before the correction. At point a, the gradient of the matrix $\boldsymbol{D}$ is given by the tensor $\mathcal{C}$ given by Eq. (24). Line l indicates the linearized

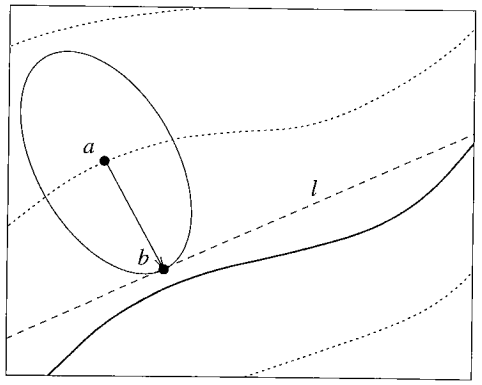


Fig. 1 Correction of the flow matrix.

constraint. The first order solution is sought on line l. The ellipse represents the contour of a constant *Mahalanobis distance* $(\Delta \boldsymbol{F}; (\mathcal{V}_F^0)_8^- \Delta \boldsymbol{F})^{1/2}$ from point a. The solution b is the point where the ellipse is in contact with line l. The next iteration begins at point b.

The procedure for the flow matrix correction is summarized as follows:

Step 1: Compute $\Delta \boldsymbol{F}$ by Eq. (27), and update $\hat{\boldsymbol{F}}$ and $\mathcal{V}_F^0$ by Eqs. (30) and (31), respectively.

Step 2: Compute $\boldsymbol{D}$ (= the left-hand side of Eq. (12)). If $\|\boldsymbol{D}\| \approx 0$, then stop. Otherwise, go back to Step 1:.

The corrected flow matrix $\hat{\boldsymbol{F}}$ is decomposed into $\hat{\boldsymbol{v}}$ and $\hat{\boldsymbol{\omega}}$ by using Eqs. (13) and (14):

$$\hat{\boldsymbol{v}} = (A[\hat{\boldsymbol{F}}]_{32}\ A[\hat{\boldsymbol{F}}]_{13}\ A[\hat{\boldsymbol{F}}]_{21})^\top, \tag{32}$$

$$\hat{\boldsymbol{\omega}} = \frac{1}{2}(\mathrm{tr}\, S[\hat{\boldsymbol{F}}])\hat{\boldsymbol{v}} - 2S[\hat{\boldsymbol{F}}]\hat{\boldsymbol{v}}. \tag{33}$$

5. Covariances of the Motion Parameters

The covariance matrices $\boldsymbol{V}_v$, $\boldsymbol{V}_\omega$ and $\boldsymbol{V}_{\omega v}$ of the computed motion parameters $\{\hat{\boldsymbol{v}}, \hat{\boldsymbol{\omega}}\}$ are given as follows (we omit the derivation [6]):

$$\begin{pmatrix} \boldsymbol{V}_v & \boldsymbol{V}_{\omega v}^\top \\ \boldsymbol{V}_{\omega v} & \boldsymbol{V}_\omega \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_v & \boldsymbol{A}_{\omega v}^\top \\ \boldsymbol{A}_{\omega v} & \boldsymbol{A}_\omega \end{pmatrix}^- . \tag{34}$$

Here,

$$\boldsymbol{A}_v = \sum_{\alpha=1}^{n} \frac{(\boldsymbol{P}_v \bar{\boldsymbol{a}}_\alpha)(\boldsymbol{P}_v \bar{\boldsymbol{a}}_\alpha)^\top}{(\boldsymbol{v} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\boldsymbol{v} \times \boldsymbol{x}_\alpha))}, \tag{35}$$

$$\boldsymbol{A}_\omega = \sum_{\alpha=1}^{n} \frac{\bar{\boldsymbol{b}}_\alpha \bar{\boldsymbol{b}}_\alpha^\top}{(\boldsymbol{v} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\boldsymbol{v} \times \boldsymbol{x}_\alpha))}, \tag{36}$$

$$\boldsymbol{A}_{\omega v} = \sum_{\alpha=1}^{n} \frac{\bar{\boldsymbol{b}}_\alpha (\boldsymbol{P}_v \bar{\boldsymbol{a}}_\alpha)^\top}{(\boldsymbol{v} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\boldsymbol{v} \times \boldsymbol{x}_\alpha))}, \tag{37}$$

$$\bar{\boldsymbol{a}}_\alpha = \boldsymbol{x}_\alpha \times \bar{\dot{\boldsymbol{x}}}_\alpha + \|\boldsymbol{x}_\alpha\|^2 \boldsymbol{\omega} - (\boldsymbol{x}_\alpha, \boldsymbol{\omega})\boldsymbol{x}_\alpha, \tag{38}$$

$$\bar{\boldsymbol{b}}_\alpha = \|\boldsymbol{x}_\alpha\|^2 \boldsymbol{v} - (\boldsymbol{x}_\alpha, \boldsymbol{v})\boldsymbol{x}_\alpha, \tag{39}$$

$$\boldsymbol{P}_v = \boldsymbol{I} - \boldsymbol{v}\boldsymbol{v}^\top. \tag{40}$$

Equation (34) is equivalent to what is known as the *Cramer-Rao lower bound* in statistics and gives a theoretical bound on the accuracy of estimation. It can also be shown that maximum likelihood estimation attains this bound in the first order. In actual computation, the motion parameters $\{\boldsymbol{v}, \boldsymbol{\omega}\}$ and the optical flow $\bar{\dot{\boldsymbol{x}}}_\alpha$ in the above equations are replaced by their estimated values (the optical flow is estimated by Eq. (43)). Since Eqs. (35)–(37) involve the covariance matrix $\boldsymbol{V}_\alpha = \epsilon^2 \boldsymbol{V}_\alpha^0$, we need to estimate the square noise level ϵ^2, for which we use the following estimator:

$$\hat{\epsilon}^2 = \frac{\hat{J}_2^0}{n-5}. \tag{41}$$

Here, $\hat{J}_2^0$ is the value of J_2 obtained by replacing $\boldsymbol{v}$, $\boldsymbol{\omega}$ and $\boldsymbol{V}_\alpha$ by $\hat{\boldsymbol{v}}$, $\hat{\boldsymbol{\omega}}$ and $\boldsymbol{V}_\alpha^0$, respectively, in Eq. (15). Equation (41) is obtained from the fact that $\hat{J}_2^0/\epsilon^2$ is a χ^2 random variable with $n-5$ degrees of freedom [7]. It follows that the expectation and the variance of $\hat{\epsilon}^2$ are

$$E[\hat{\epsilon}^2] = \epsilon^2, \quad V[\hat{\epsilon}^2] = \frac{2\epsilon^4}{n-5}. \tag{42}$$

6. Depth and Its Variance

The next step is reconstruction of the depth map. Firstly, we estimate the true optical flow $\bar{\dot{\boldsymbol{x}}}_\alpha$. In Sect. 2, we eliminated $\bar{\dot{\boldsymbol{x}}}_\alpha$ to obtain function J_2 from function J_1. The true optical flow is estimated by computing the eliminated parameter $\bar{\dot{\boldsymbol{x}}}_\alpha$. Since $\hat{\boldsymbol{v}}$ and $\hat{\boldsymbol{\omega}}$ have already been determined, minimizing J_1 under the epipolar constraint (1) reduces to minimization of a quadric form. The solution $\hat{\dot{\boldsymbol{x}}}_\alpha$ is given as follows:

$$\hat{\dot{\boldsymbol{x}}}_\alpha = \dot{\boldsymbol{x}}_\alpha - \frac{\hat{e}_\alpha \boldsymbol{V}_\alpha^0(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha)}{(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha^0(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))}, \tag{43}$$

$$\hat{e}_\alpha = |\boldsymbol{x}_\alpha, \dot{\boldsymbol{x}}_\alpha + \hat{\boldsymbol{v}} + \hat{\boldsymbol{\omega}} \times \boldsymbol{x}_\alpha, \hat{\boldsymbol{v}}|. \tag{44}$$

The Z coordinate at the αth point, which we denote by $\hat{Z}_\alpha$, is computed from $\hat{\dot{\boldsymbol{x}}}_\alpha$ in the form

$$\hat{Z}_\alpha = -\frac{(\hat{\boldsymbol{v}}, \boldsymbol{Q}_\alpha^\top (\boldsymbol{V}_\alpha^0)_2^- \boldsymbol{Q}_\alpha \hat{\boldsymbol{v}})}{(\hat{\boldsymbol{v}}, \boldsymbol{Q}_\alpha^\top (\boldsymbol{V}_\alpha^0)_2^- \boldsymbol{Q}_\alpha (\hat{\dot{\boldsymbol{x}}}_\alpha + \hat{\boldsymbol{\omega}} \times \boldsymbol{x}_\alpha))}, \tag{45}$$

where $\boldsymbol{Q}_\alpha = \boldsymbol{I} - \boldsymbol{x}_\alpha \boldsymbol{k}^\top$ and $\boldsymbol{k} = (0\,0\,1)^\top$.

We now estimate the variance of $\hat{Z}_\alpha$. Since $\hat{Z}_\alpha$ is computed from $\hat{\dot{\boldsymbol{x}}}_\alpha$, which is computed from the observed optical flow $\dot{\boldsymbol{x}}_\alpha$, it appears that the variance of $\hat{Z}_\alpha$ could be computed by propagating the error in $\dot{\boldsymbol{x}}_\alpha$ through the computation steps for $\hat{Z}_\alpha$. However, $\hat{Z}_\alpha$ also depends upon the motion parameters $\{\hat{\boldsymbol{v}}, \hat{\boldsymbol{\omega}}\}$, which are estimated from the same data $\dot{\boldsymbol{x}}_\alpha$. If errors in the motion parameters and errors in the optical flow are considered at the same time, the analysis becomes too complicated. As a compromise, we evaluate the effect of errors in $\{\hat{\boldsymbol{v}}, \hat{\boldsymbol{\omega}}\}$ and that of errors in $\dot{\boldsymbol{x}}_\alpha$ separately and then superimpose the two effects.

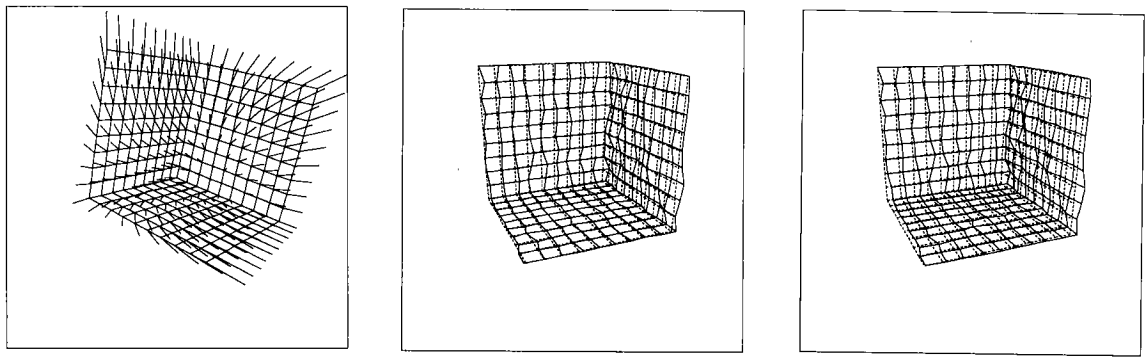
In order to evaluate the effect of errors in the flow, we temporarily assume that $\{\hat{\boldsymbol{v}}, \hat{\boldsymbol{\omega}}\}$ are the true parameters. Then, the covariance matrix of the estimated flow $\hat{\dot{\boldsymbol{x}}}_\alpha$, which we denote by $\boldsymbol{V}_{\dot{x}\alpha}^{(f)}$, is given by

$$\boldsymbol{V}_{\dot{x}\alpha}^{(f)} = \boldsymbol{V}_\alpha - \frac{(\boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))(\boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))^\top}{(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))}. \tag{46}$$

The variance of Z_α, which we denote by $V_{Z\alpha}^{(f)}$, is given by

$$V_{Z\alpha}^{(f)} = \frac{\hat{Z}_\alpha^4 (\mathrm{tr} \boldsymbol{V}_{\dot{x}\alpha}^{(f)})}{(\hat{\boldsymbol{v}}, \boldsymbol{S}_\alpha \hat{\boldsymbol{v}})}, \tag{47}$$

$$\boldsymbol{S}_\alpha = \boldsymbol{x}_\alpha \times \mathrm{diag}(1,1,0) \times \boldsymbol{x}_\alpha, \tag{48}$$



(a) object shape and generated optical flow.

(b) reconstructed shape (present method).

(c) reconstructed shape (previous method).

Fig. 2 Generated optical flow and reconstructed shapes.

where $\mathrm{diag}(1,1,0)$ is a diagonal matrix whose diagonal elements are 1, 1 and 0 in that order (see Appendix for the product rule for vectors and matrices).

We next evaluate the effect of errors in the motion parameters. This time, the estimated flow $\hat{\dot{\boldsymbol{x}}}_\alpha$ is regarded as true. First, compute the following covariance matrices (see Eq. (34)):

$$V_{e\alpha} = (\hat{\boldsymbol{a}}_\alpha, \boldsymbol{V}_v\hat{\boldsymbol{a}}_\alpha) + (\hat{\boldsymbol{b}}_\alpha, \boldsymbol{V}_\omega\hat{\boldsymbol{b}}_\alpha) + 2(\hat{\boldsymbol{b}}_\alpha, \boldsymbol{V}_{\omega v}\hat{\boldsymbol{a}}_\alpha), \tag{49}$$

$$\boldsymbol{V}_{ve\alpha} = \boldsymbol{V}_v\hat{\boldsymbol{a}}_\alpha + \boldsymbol{V}_{\omega v}^\top\hat{\boldsymbol{b}}_\alpha, \tag{50}$$

$$\boldsymbol{V}_{\omega e\alpha} = \boldsymbol{V}_{\omega v}\hat{\boldsymbol{a}}_\alpha + \boldsymbol{V}_v\hat{\boldsymbol{b}}_\alpha. \tag{51}$$

Here, we have defined

$$\hat{\boldsymbol{a}}_\alpha = \boldsymbol{x}_\alpha \times \hat{\dot{\boldsymbol{x}}}_\alpha + \|\boldsymbol{x}_\alpha\|^2\hat{\boldsymbol{\omega}} - (\boldsymbol{x}_\alpha, \hat{\boldsymbol{\omega}})\boldsymbol{x}_\alpha, \tag{52}$$

$$\hat{\boldsymbol{b}}_\alpha = \|\boldsymbol{x}_\alpha\|^2\hat{\boldsymbol{v}} - (\boldsymbol{x}_\alpha, \hat{\boldsymbol{v}})\boldsymbol{x}_\alpha. \tag{53}$$

Next, compute the following covariance matrices:

$$\boldsymbol{V}_{\dot{x}\alpha}^{(m)} = \frac{V_{e\alpha}(\boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))(\boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))^\top}{(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))^2}, \tag{54}$$

$$\boldsymbol{V}_{v\dot{x}\alpha} = \frac{\boldsymbol{V}_{ve\alpha}(\boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))^\top}{(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))}, \tag{55}$$

$$\boldsymbol{V}_{\omega\dot{x}\alpha} = \frac{\boldsymbol{V}_{\omega e\alpha}(\boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))^\top}{(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha, \boldsymbol{V}_\alpha(\hat{\boldsymbol{v}} \times \boldsymbol{x}_\alpha))}. \tag{56}$$

The variance of $\hat{Z}_\alpha$, which we denote by $V_{Z\alpha}^{(m)}$, is given by

$$V_{Z\alpha}^{(m)} = \frac{\hat{Z}_\alpha^4}{(\hat{\boldsymbol{v}}, \boldsymbol{S}_\alpha\hat{\boldsymbol{v}})}\left((\boldsymbol{V}_{\dot{x}\alpha}^{(m)}; \boldsymbol{S}_\alpha) + \frac{(\boldsymbol{V}_v; \boldsymbol{S}_\alpha)}{\hat{Z}_\alpha^2} + (\boldsymbol{x}_\alpha \times \boldsymbol{V}_\omega \times \boldsymbol{x}_\alpha; \boldsymbol{S}_\alpha) + \frac{2(\boldsymbol{V}_{v\dot{x}\alpha}; \boldsymbol{S}_\alpha)}{\hat{Z}_\alpha} - 2(\boldsymbol{x}_\alpha \times \boldsymbol{V}_{\omega\dot{x}}; \boldsymbol{S}_\alpha) - \frac{2(\boldsymbol{x}_\alpha \times \boldsymbol{V}_{\omega v}; \boldsymbol{S}_\alpha)}{\hat{Z}_\alpha}\right), \tag{57}$$

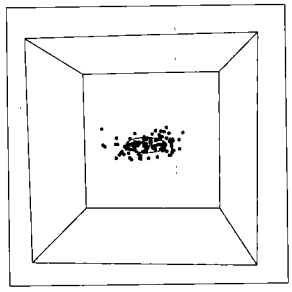
where $\boldsymbol{S}_\alpha$ is defined by Eq. (48). The total variance $V_{Z\alpha}$ of $\hat{Z}_\alpha$ is given as the sum of $V_{Z\alpha}^{(f)}$ and $V_{Z\alpha}^{(m)}$:

$$V_{Z\alpha} = V_{Z\alpha}^{(f)} + V_{Z\alpha}^{(m)}. \tag{58}$$

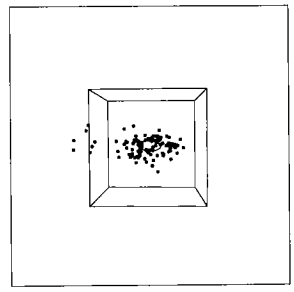
7. Experiments

We conducted experiments using computer generated data. Figure 2(a) shows the object shape and the generated optical flow. The focal length is 600 pixels, and the image size is 512×512 pixels. The distance between the object and the camera is approximately 6×10^5 pixels. The motion parameters are $\boldsymbol{v} = (0, -1.15, 1.15)^\top \times 10^5$ pixels/frame and $\boldsymbol{\omega} = (-0.21, 0, 0)^\top$ rad/frame. Random Gaussian noise of mean 0 and variance 1 pixel was added to the x- and y-components of the flow independently. Figure 2(b) shows the reconstructed object shape (the true shape is superimposed in dotted lines). For comparison, the coresponding result obtained by our previous method [5] is shown in Fig. 2(c). It is observed that the optimal correction of the flow matrix has improved the accuracy. This simulation was repeated 100 times with different series of random numbers. The estimated motion parameters are plotted in three dimensions in Fig. 3. The cubic frames, which are shown merely as a reference, have the same absolute size and are centered at the true values. Figures 3(a) and 3(c) are for the present method; Figures 3(b) and 3(d) are for the previous method. The ellipses in the figures indicate the theoretical lower bound on the standerd diviation [6]. The distribution of the motion parameters computed by the present method is more concentrated around the true value than for the previos method and is comparable with the theoretical lower bound.

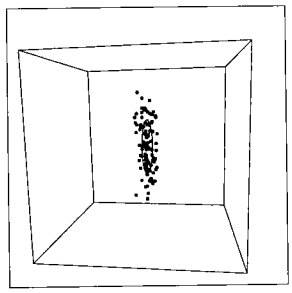
Figure 4 shows experiments using real images. Figure 4(a) shows the image we used and the optical flow detected by the method described in [8]. Figure 4(b) shows the reconstructed depth map (those points whose reliability is smaller than a threshold are not displayed). Here, the intensity corresponds to the depth: bright points have large depths. Since no smoothing operation is applied, this depth map is inhomogenious, but a



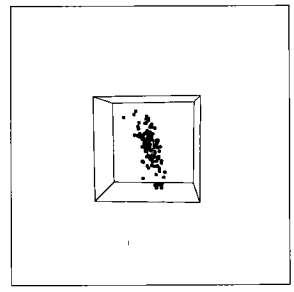
(a) translation velocity (present method).



(b) translation velocity (previous method).

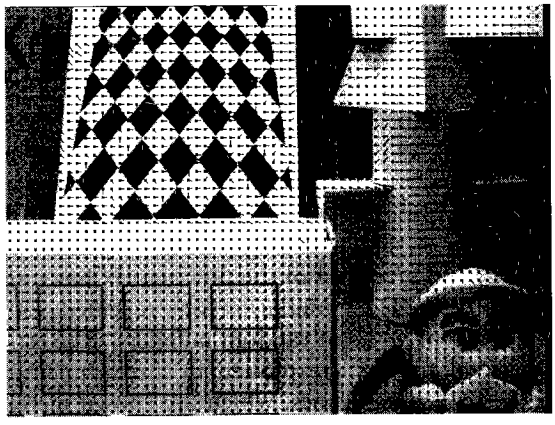


(c) rotation velocity (present method).



(d) rotation velocity (previous method).

Fig. 3 Distribution of the motion parameters.



(a) detected optical flow.

(b) depth map.

Fig. 4 Real image experiment.

global shape of the objects can be obtained. The motion parameters are estimated to be $\hat{\boldsymbol{v}} = (0.12, 0.63, 0.76)^\top$ and $\hat{\boldsymbol{\omega}} = (7.1, -0.22, -1.4)^\top \times 10^{-3}$, which seem reasonably good.

8. Conclusions

We have proposed a new structure-from-motion algorithm. The significance of our method is summarized as follows:

- The accuracy almost attains the theoretical bound if the linearization technique, the renormalization scheme and the optimal correction of the flow matrix are combined.
- Along with optimal estimates of the motion parameters and the depths, their reliability is also computed in quantitative terms.

With these advantages, our method is expected to play an important

Acknowledgments

The authors thank Mr. Hiroyuki Morishita of Gunma

University for conducting the experiments together. This work was in part supported by the Ministry of Education, Science, Sports and Culture, Japan and under a Grant in Aid for Scientific Research B (No. 07458067) and the Okawa Institute of Information and Telecommunication.

References

[1] G. Adiv, "Determining three-dimensional motion and structure from optical flow generated by several moving objects," IEEE Trans. Patt. Anal. Mach. Intell., vol.7, pp.384–401, 1985.
[2] K. Daniilidis and H.H. Nagel, "Analytical results on error sensitivity of motion estimation from two views," Image Vis. Comput., vol.8, pp.287–303, 1990.
[3] D.J. Heeger and A.D. Jepson, "Subspace methods for recovering rigid motion I: algorithm and interpretation," Intern. J. Comput. Vis., vol.7, pp.95–117, 1992.
[4] K. Kanatani, "Geometric Computation for Machine Vision," Oxford University Press, Oxford, 1993.
[5] K. Kanatani, "3-D interpretation of optical flow by renormalization," Intern. J. Comput. Vis., vol.11, no.3, pp.267–282, 1993.
[6] K. Kanatani, "Accuracy bound of parametric fitting," IPSJ Technical Report, vol.94-CV-91-3, pp.15–22, 1994 (in Japanese).
[7] K. Kanatani, "Statistical Optimization for Geometric Computation: Theory and Practice," Elsevier Science, Amsterdam, 1996.
[8] N. Ohta, "Image movement detection with reliability indices," IEICE Trans., vol.E74, no.10, pp.3379–3388, Oct. 1991.
[9] N. Ohta, "Structure from motion with confidence measure and its application for moving object detection," IEICE Trans., vol.J76-D-II, no.8, pp.1562–1571, 1993 (in Japanese).
[10] N. Tagawa, T. Toriu, and T. Endoh, "Un-biased linear algorithm for recovering three-dimensional motion from optical flow," IEICE Trans. Infor. Syst., vol.E76-D, pp.1263–1275, 1993.
[11] S. Ullman, "The Interpretation of Visual Motion," MIT Press, Cambridge MA, 1979.
[12] J. Weng, T.S. Huang, and N. Ahuja, "Motion and structure from two perspective views: algorithms, error analysis and error estimation," IEEE Trans. Patt. Anal. Mach. Intell., vol.11, pp.451–467, 1989.
[13] X. Zhuang, T.S. Huang, N. Ahuja, and R.M. Haralick, "A simplified linear optical flow-motion algorithm," Comput. Vis. Graph. Image Process., vol.42, pp.334–344, 1988.

Appendix: Notations and Definitions

Operations

The following lists notations of operations on tensors, matrices and vectors used in this paper.

- The *product* of a tensor $\mathcal{T}=(\mathcal{T}_{ijkl})$ and a matrix $\boldsymbol{A}=(A_{ij})$:

$$(\mathcal{T}\boldsymbol{A})_{ij} = \sum_{k,l=1}^{3} \mathcal{T}_{ijkl}A_{kl}\,.$$

Here, $(\mathcal{T}\boldsymbol{A})_{ij}$ denotes the (ij) element of matrix $\mathcal{T}\boldsymbol{A}$.

- The *exterior product* of a vector $\boldsymbol{v}$ and a matrix $\boldsymbol{A}=(\boldsymbol{a}_1\ \boldsymbol{a}_2\ \boldsymbol{a}_3)$ ($\boldsymbol{a}_i$ is the i-th column of $\boldsymbol{A}$):

$$\boldsymbol{v}\times\boldsymbol{A} = (\boldsymbol{v}\times\boldsymbol{a}_1\ \ \boldsymbol{v}\times\boldsymbol{a}_2\ \ \boldsymbol{v}\times\boldsymbol{a}_3),$$
$$\boldsymbol{A}\times\boldsymbol{v} = (\boldsymbol{v}\times\boldsymbol{A}^\top)^\top.$$

- The *inner product* of matrices $\boldsymbol{A}=(A_{ij})$ and $\boldsymbol{B}=(B_{ij})$ and the norm of matrix $\boldsymbol{A}=(A_{ij})$:

$$(\boldsymbol{A};\boldsymbol{B}) = \sum_{i,j=1}^{3} A_{ij}B_{ij}\,,$$

$$\|\boldsymbol{A}\| = \sqrt{(\boldsymbol{A};\boldsymbol{A})} = \sqrt{\sum_{i,j=1}^{3} A_{ij}^2}\,.$$

- The *symmetrization operator* $S[\cdot]$ and the *anti-symmetrization operator* $A[\cdot]$ on matrix $\boldsymbol{A}$:

$$S[\boldsymbol{A}] = \frac{1}{2}(\boldsymbol{A}+\boldsymbol{A}^\top),\quad A[\boldsymbol{A}] = \frac{1}{2}(\boldsymbol{A}-\boldsymbol{A}^\top).$$

- The *scalar triplet product* of vectors $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{w}$:

$$|\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}| = (\boldsymbol{u}\times\boldsymbol{v},\boldsymbol{w}) = (\boldsymbol{v}\times\boldsymbol{w},\boldsymbol{u}) = (\boldsymbol{w}\times\boldsymbol{u},\boldsymbol{v})\,.$$

Eigenvalues and Eigenmatrices

A three-dimensional tensor $\mathcal{A}=(\mathcal{A}_{abcd})$ is rearranged into a nine-dimensional matrix $\boldsymbol{A}=(A_{ij})$ by the rule described below, where div and mod denote integer division and integer remainder, respectively:

$$a = (i-1)\,\mathrm{div}\,3+1\,, \qquad \text{(A·1)}$$
$$b = (i-1)\bmod 3+1\,, \qquad \text{(A·2)}$$
$$c = (j-1)\,\mathrm{div}\,3+1\,, \qquad \text{(A·3)}$$
$$d = (j-1)\bmod 3+1\,. \qquad \text{(A·4)}$$

Let $\lambda^{(n)}$ and $\boldsymbol{u}^{(n)}$ be the nth eigenvalue and the corresponding eigenvector of $\boldsymbol{A}$, respectively. The nth *eigenvalue* of tensor $\mathcal{A}$ is also $\lambda^{(n)}$. The corresponding *eigenmatrix* $\boldsymbol{U}^{(n)}=(U_{ab}^{(n)})$ is obtained from $\boldsymbol{u}^{(n)}=(u_i^{(n)})$ by rearranging their elements according to Eqs. (A·1) and (A·2).

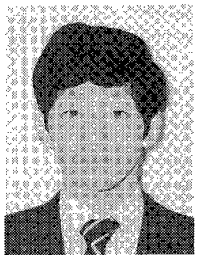
Rank-Constrained Generalized Inverse

Let $\boldsymbol{A}$ be a positive semi-definite matrix of rank r. Let $\lambda^{(n)}$ and $\boldsymbol{u}^{(n)}$ be the nth eigenvalue and the corresponding eigenvector of $\boldsymbol{A}$, respectively. The eigenvalues are indexed in descending order. For $r' \leqq r$, the *rank-constrained generalized inverse* of $\boldsymbol{A}$ is defined as follows:

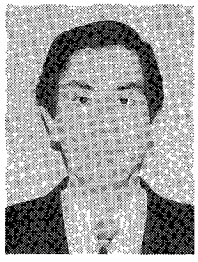
$$(\boldsymbol{A})_{r'}^{-} = \sum_{n=1}^{r'} \frac{1}{\lambda^{(n)}} \boldsymbol{u}^{(n)} \boldsymbol{u}^{(n)\top}. \qquad (\text{A}\cdot 5)$$

If $r = r'$, the rank-constrained generalized inverse $(\boldsymbol{A})_{r'}^{-}$ coincides with the generalized inverse $\boldsymbol{A}^{-}$.

The rank-constrained generalized inverse of a three-dimensional tensor $\mathcal{A}$ can also be defined. Namely, tensor $\mathcal{A}$ can be converted into a nine-dimensional matrix $\boldsymbol{A}$ as stated in Appendix. If $\boldsymbol{A}$ is positive semi-definite, the rank-constrained generalized inverse $(\boldsymbol{A})_{r'}^{-}$ can be computed by Eq. (A· 5). The rank-constrained generalized inverse $(\mathcal{A})_{r'}^{-}$ of tensor $\mathcal{A}$ is obtained from $(\boldsymbol{A})_{r'}^{-}$ by applying the conversion rule of Eqs. (A· 1), (A· 2), (A· 3) and (A· 4).



Naoya Ohta received his ME degree from the Tokyo Institute of Technology in 1985. He engaged in research and development of image processing systems at the Pattern Recognition Research Laboratory of NEC. He is currently Assistant Professor of computer science at Gunma University. He was a research affiliate of the Media Laboratory in MIT from 1991 to 1992.



Kenichi Kanatani received his Ph.D. in applied mathematics from the University of Tokyo in 1979. He is currently Professor of computer science at Gunma University. He is the author of *Group-Theoretical Methods in Image Understanding* (Springer, 1990) and *Geometric Computation for Machine Vision* (Oxford University Press, 1993).