PAPER

3-D Motion Analysis of a Planar Surface by Renormalization

Kenichi KANATANI†, ***Member and*** **Sachio TAKEDA**††, ***Nonmember***

SUMMARY This paper presents a *theoretically best* algorithm within the framework of our image noise model for reconstructing 3-D from two views when all the feature points are on a planar surface. Pointing out that statistical bias is introduced if the least-squares scheme is used in the presence of image noise, we propose a scheme called *renormalization*, which automatically removes statistical bias. We also present an optimal correction scheme for canceling the effect of image noise in individual feature points. Finally, we show numerical simulation and confirm the effectiveness of our method.

key words: *computer vision, 3-D motion analysis, planar surface, projective transformation, renormalization, statistical bias*

1. Introduction

3-D motion analysis from two images, known as *structure from motion*, is one of the most fundamental problems of computer vision, and many algorithms have been studied for this problem [1]–[3], [6], [9]. However, usual 3-D reconstruction algorithms fail if all the feature points are on a planar surface in the scene. Algorithms for coplanar feature points have also been proposed in various forms [3], [7], [9]; they are all based on the following principle.

If a 3-D motion of a planar surface is observed by a camera, the resulting 2-D image motion is a *projective transformation*, which is specified by a 3×3 matrix called the *transformation matrix* [3], [8]. Once the transformation matrix is computed from the observed image motion, it can be easily decomposed into the parameters of the planar surface and the parameters of the camera motion [3], [7], [9]. Hence, the 3-D motion analysis consists of the following two stages:

1. computing the transformation matrix from the observed image motion;
2. decomposing the computed transformation matrix into surface and motion parameters.

In the past, the second stage has been intensively studied [3], [7], [9], but not much attention has been paid to the first stage; only the use of the least-squares fitting scheme has been suggested. Also, it has been widely thought that the goal of the analysis is to determine the camera motion and *reconstruct the planar surface*. In real circumstances, however, reconstructing *individual feature points* is also important. The aim of this paper is as follows:

- We point out that *statistical bias* is introduced if the least-squares scheme is used in the first stage in the presence of image noise.
- We propose a scheme called *renormalization* for computing the transformation matrix *in an optimal way.*
- We present an *optimal correction scheme* for canceling the effect of image noise so that the 3-D positions of *individual feature points* are reconstructed *in an optimal way.*
- We show numerical simulation by using random noise and confirm the effectiveness of our method.

2. 3-D Analysis of Planar Surface Motion

The camera is associated with an XYZ coordinate system with origin O at the center of the lens and Z-axis along the optical axis. The plane $Z = 1$ is identified with the image plane, on which an xy image coordinate system is defined around the Z-axis such that the x- and y-axes are parallel to the X- and Y-axes, respectively. A point on the image plane with image coordinates (x, y) is represented by its *position vector* $\boldsymbol{x} = (x, y, 1)^\top$, where the superscript $\top$ denotes transpose.

Consider a planar surface in the scene. Let $\boldsymbol{n}$ be its unit surface normal, and d its distance (positive in the direction of $\boldsymbol{n}$) from the origin O: if we put $\boldsymbol{r} = (X, Y, Z)^\top$, the equation of the surface is $(\boldsymbol{n}, \boldsymbol{r}) = d$. In this paper, we write $(\boldsymbol{a}, \boldsymbol{b})$ to denote the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$. We call $\{\boldsymbol{n}, d\}$ the *surface parameters.*

Since an object motion relative to a stationary camera is equivalent to a camera motion relative to a stationary object, we assume that the camera moves in a stationary scene. Displacing one camera is equivalent to positioning two cameras in the scene, so we consider two cameras positioned in such a way that the second camera is translated from the first camera position by $\boldsymbol{h}$ and rotated around the center of the lens by $\boldsymbol{R}$ (Fig. 1): we call $\{\boldsymbol{h}, \boldsymbol{R}\}$ the *motion parameters.*

Manuscript received January 13, 1995.
†The author is with the Department of Computer Science, Gunma University, Kiryu-shi, 376 Japan.
††The author is with NTT, Tokyo, 100 Japan.

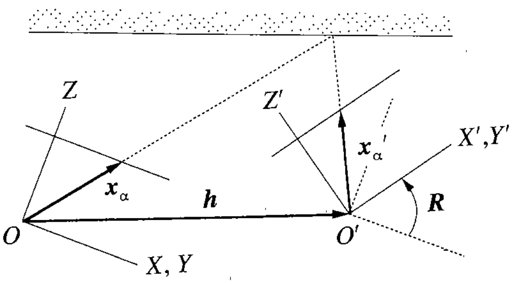


Fig. 1 Camera motion.

Suppose we observe N feature points in the scene. Let $\bar{\boldsymbol{x}}_\alpha$ and $\bar{\boldsymbol{x}}'_\alpha$ be the position vectors of the αth feature point before and after the camera motion, respectively, in the absence of image noise. The necessary and sufficient condition that the feature point is on the planar surface is

$$\bar{\boldsymbol{x}}'_\alpha \times \boldsymbol{A}\bar{\boldsymbol{x}}_\alpha = \boldsymbol{0}, \tag{1}$$

where $\boldsymbol{A}$ is a matrix given as follows [1], [3], [7]–[9]:

$$\boldsymbol{A} = \boldsymbol{R}^\top(\boldsymbol{h}\boldsymbol{n}^\top - d\boldsymbol{I}). \tag{2}$$

Here, $\boldsymbol{I}$ denotes the unit matrix. In the presence of noise, the observed position vectors $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ do not necessarily satisfy Eq. (1). Hence, the 3-D motion analysis reduces to the following two subproblems:

Problem 1: Estimate the transformation matrix $\boldsymbol{A}$ that satisfies Eq. (1) for $\alpha = 1, ..., N$ from the data $\{\boldsymbol{x}_\alpha\}$ and $\{\boldsymbol{x}'_\alpha\}$.

Problem 2: From the computed transformation matrix $\boldsymbol{A}$, compute the surface and motion parameters $\{\boldsymbol{n}, d\}$ and $\{\boldsymbol{h}, \boldsymbol{R}\}$ that satisfy Eq. (2).

Here, we note the following two facts:

1. As will be shown shortly, *any* nonsingular matrix $\boldsymbol{A}$ can be decomposed into $\{\boldsymbol{h}, \boldsymbol{R}\}$ and $\{\boldsymbol{n}, d\}$ (although not uniquely). Hence, the transformation matrix $\boldsymbol{A}$ in Problem 1 is *unconstrained*.

2. Equation (1) implies that the transformation matrix $\boldsymbol{A}$ is determined only *up to scale*. As a result, the scale of the the translation $\boldsymbol{h}$ and the distance d is indeterminate. This corresponds to the well known fact that given an image motion a large camera motion relative to a large object in the distance is indistinguishable from a small camera motion relative to a small object nearby.

3. Estimation of the Transformation Matrix

Writing $\boldsymbol{x}_\alpha = \bar{\boldsymbol{x}}_\alpha + \Delta\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha = \bar{\boldsymbol{x}}'_\alpha + \Delta\boldsymbol{x}'_\alpha$, we regard the noise terms $\Delta\boldsymbol{x}_\alpha$ and $\Delta\boldsymbol{x}'_\alpha$ as independent random variables and define their *covariance matrices* by $V[\boldsymbol{x}_\alpha] = E[\Delta\boldsymbol{x}_\alpha\Delta\boldsymbol{x}_\alpha^\top]$ and $V[\boldsymbol{x}'_\alpha] = E[\Delta\boldsymbol{x}'_\alpha\Delta\boldsymbol{x}'^\top_\alpha]$, where the symbol $E[\,\cdot\,]$ denotes expectation. These matrices are singular and in general have rank 2. The noise distribution is in general different from point to point.

In practice, it is very difficult to estimate the covariance matrix of each feature point exactly. However, it is often easy to predict the qualitative properties of noise characteristics such as uniformity and isotropy. In view of this, we assume that the covariance matrices are known only *up to scale*. In other words, we write

$$V[\boldsymbol{x}_\alpha] = \epsilon^2 V_0[\boldsymbol{x}_\alpha], \qquad V[\boldsymbol{x}'_\alpha] = \epsilon^2 V_0[\boldsymbol{x}'_\alpha], \tag{3}$$

and assume that the matrices $V_0[\boldsymbol{x}_\alpha]$ and $V_0[\boldsymbol{x}'_\alpha]$ are known while the constant ϵ is unknown. We call $V_0[\boldsymbol{x}_\alpha]$ and $V_0[\boldsymbol{x}'_\alpha]$ the *normalized covariance matrices* and ϵ the *noise level*.

If the image noise is Gaussian and the product of image terms are approximated by a Gaussian random variable, a statistically optimal estimator of the transformation matrix $\boldsymbol{A}$ is obtained by minimizing the following function [4]:

$$J[\boldsymbol{A}] = \sum_{\alpha=1}^{N}(\boldsymbol{x}'_\alpha \times \boldsymbol{A}\boldsymbol{x}_\alpha, \boldsymbol{W}_\alpha(\boldsymbol{A})(\boldsymbol{x}'_\alpha \times \boldsymbol{A}\boldsymbol{x}_\alpha)). \tag{4}$$

The matrix $\boldsymbol{W}_\alpha(\boldsymbol{A})$ is defined by

$$\begin{aligned}\boldsymbol{W}_\alpha(\boldsymbol{A}) = \Big(&\boldsymbol{x}'_\alpha \times \boldsymbol{A}V_0[\boldsymbol{x}_\alpha]\boldsymbol{A}^\top \times \boldsymbol{x}'_\alpha \\ &+(\boldsymbol{A}\boldsymbol{x}_\alpha) \times V_0[\boldsymbol{x}'_\alpha] \times (\boldsymbol{A}\boldsymbol{x}_\alpha) \\ &+\epsilon^2[\boldsymbol{A}V_0[\boldsymbol{x}_\alpha]\boldsymbol{A}^\top \times V_0[\boldsymbol{x}'_\alpha]]\Big)_2^-,\end{aligned} \tag{5}$$

where the symbol $(\cdots)_2^-$ denotes the (Moore-Penrose) generalized inverse whose rank is constrained to be 2 (i.e., obtained by replacing the smallest eigenvalue by 0 in the spectral decomposition [4]). For vectors $\boldsymbol{a} = (a_i)$ and $\boldsymbol{b} = (b_i)$ and matrices $\boldsymbol{U} = (U_{ij})$ and $\boldsymbol{V} = (V_{ij})$, we write $\boldsymbol{a} \times \boldsymbol{U} \times \boldsymbol{b}$ and $[\boldsymbol{U} \times \boldsymbol{V}]$ to denote the matrices whose (ij) elements are $\sum_{k,l,m,n=1}^{3} \epsilon_{ikl}\epsilon_{jmn}a_k b_m U_{ln}$ and $\sum_{k,l,m,n=1}^{3} \epsilon_{ikl}\epsilon_{jmn}U_{km}V_{ln}$, respectively, where ϵ_{ijk} is the *Eddington epsilon*, taking values 1 and -1 if (ijk) is an even and odd permutation of (123), respectively, and taking value 0 otherwise.

It is easily seen from Eq. (5) that $J[c\boldsymbol{A}] = J[\boldsymbol{A}]$ for an arbitrary nonzero constant c. Hence, we can arbitrarily normalize $\boldsymbol{A}$ for removing the scale indeterminacy. For simplicity, we adopt the normalization $\|\boldsymbol{A}\| = 1$, where the norm of a matrix $\boldsymbol{U} = (U_{ij})$ is defined by $\|\boldsymbol{U}\| = \sqrt{\sum_{i,j=1}^{3} U_{ij}{}^2}$.

4. Statistical Bias

If $\boldsymbol{W}_\alpha(\boldsymbol{A})$ is replaced by a constant matrix $\boldsymbol{W}_\alpha$ in Eq. (4), the optimization takes the following form:

$$\tilde{J}[\boldsymbol{A}] = (\boldsymbol{A}; \mathcal{M}\boldsymbol{A}) \to \min. \tag{6}$$

Here, we define the inner product of matrices $\boldsymbol{U} = (U_{ij})$ and $\boldsymbol{V} = (V_{ij})$ by $(\boldsymbol{U}; \boldsymbol{V}) = \sum_{i,j=1}^{3} U_{ij}V_{ij}$. In the above equation, $\mathcal{M}$ is a tensor defined by

$$\mathcal{M} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{k,l=1}^{3} W_\alpha^{(kl)}(\boldsymbol{e}^{(k)} \times \boldsymbol{x}'_\alpha \boldsymbol{x}_\alpha^\top) \otimes(\boldsymbol{e}^{(l)} \times \boldsymbol{x}'_\alpha \boldsymbol{x}_\alpha^\top), \tag{7}$$

where $W_\alpha^{(kl)}$ is the (kl) element of matrix $\boldsymbol{W}_\alpha$. We call $\mathcal{M}$ the *moment tensor*. In the above equation, $\boldsymbol{e}^{(1)} = (1,0,0)^\top$, $\boldsymbol{e}^{(2)} = (0,1,0)^\top$, and $\boldsymbol{e}^{(3)} = (0,0,1)^\top$, and the symbol $\otimes$ denotes tensor product. For a tensor $\mathcal{T} = (T_{ijkl})$ and a matrix $\boldsymbol{U} = (U_{ij})$, we define the action $\mathcal{T}\boldsymbol{U}$ of tensor $\mathcal{T}$ to matrix $\boldsymbol{U}$ to be the matrix whose (ij) element is $\sum_{k,l=1}^{3} T_{ijkl}U_{kl}$. If $\mathcal{T}\boldsymbol{U} = \lambda \boldsymbol{U}$ for a scalar λ, we say that λ is the *eigenvalue* of tensor $\mathcal{T}$ for the *eigenmatrix* $\boldsymbol{U}$.

The solution to the minimization (6) under the constraint $\|\boldsymbol{A}\| = 1$ is given by an eigenmatrix of norm 1 for the smallest eigenvalue of the moment tensor $\mathcal{M}$ [3]. Eigenvalues and eigenmatrices of $\mathcal{M}$ are computed by regarding $\mathcal{M}$ as a 9×9 matrix by rearranging its elements and computing its eigenvalues and eigenvectors [3]. We call (6) the *least-squares approximation* with *weights* $W_\alpha^{(kl)}$.

From this observation, it appears that the solution that minimizes Eq. (4) is obtained by the following procedure. First, we guess an initial estimate $\boldsymbol{A}_0$. Replacing $\boldsymbol{W}_\alpha(\boldsymbol{A})$ by $\boldsymbol{W}_\alpha(\boldsymbol{A}_0)$, we obtain an updated solution $\boldsymbol{A}_1$. Replacing $\boldsymbol{W}_\alpha(\boldsymbol{A})$ by $\boldsymbol{W}_\alpha(\boldsymbol{A}_1)$, we repeat the same computation and iterate this process. However, this process introduces statistical bias into the solution. This is reasoned as follows.

Let $\bar{\mathcal{M}}$ be the unperturbed moment tensor obtained by replacing $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ by $\bar{\boldsymbol{x}}_\alpha$ and $\bar{\boldsymbol{x}}'_\alpha$, respectively, in Eq. (7). Equation (1) implies that $\bar{\mathcal{M}}\boldsymbol{A} = \boldsymbol{O}$, i.e., $\boldsymbol{A}$ is the eigenmatrix of tensor $\bar{\mathcal{M}}$ for eigenvalue 0. From Eq. (7), we can easily see that $E[\mathcal{M}] = \bar{\mathcal{M}} + O(\epsilon^2)$. Hence, the expectation of the computed eigenmatrix of $\mathcal{M}$ is perturbed from $\boldsymbol{A}$ by $O(\epsilon^2)$ according to the *perturbation theorem* [3].

5. Renormalization

By a detailed analysis, we can evaluate the exact amount of the statistical bias of $\mathcal{M}$ and *subtract* it from $\mathcal{M}$ in the form

$$\hat{\mathcal{M}} = \mathcal{M} - \epsilon^2\mathcal{N}^{(1)} + \epsilon^4\mathcal{N}^{(2)}, \tag{8}$$

where tensors $\mathcal{N}^{(1)} = (N_{ijkl}^{(1)})$ and $\mathcal{N}^{(2)} = (N_{ijkl}^{(2)})$ are defined by

$$N_{ijkl}^{(1)} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{m,n,p,q=1}^{3} \epsilon_{imp}\epsilon_{knq}W_\alpha^{(mn)} \Big(V_0[\boldsymbol{x}_\alpha]_{jl}x'_{\alpha(p)}x'_{\alpha(q)} + V_0[\boldsymbol{x}'_\alpha]_{pq}x_{\alpha(j)}x_{\alpha(l)}\Big), \tag{9}$$

$$N_{ijkl}^{(2)} = \frac{1}{N}\sum_{\alpha=1}^{N}\sum_{m,n,p,q=1}^{3} \epsilon_{imp}\epsilon_{knq}W_\alpha^{(mn)} V_0[\boldsymbol{x}_\alpha]_{jl}V_0[\boldsymbol{x}'_\alpha]_{pq}. \tag{10}$$

Here, $\boldsymbol{x}_{\alpha(i)}$ and $\boldsymbol{x}'_{\alpha(i)}$ are the ith components of $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$, respectively, and $V_0[\boldsymbol{x}_\alpha]_{ij}$ and $V_0[\boldsymbol{x}'_\alpha]_{ij}$ are the (ij) elements of $V_0[\boldsymbol{x}_\alpha]$ and $V_0[\boldsymbol{x}'_\alpha]$, respectively. It can be shown that $E[\hat{\mathcal{M}}] = \bar{\mathcal{M}}$. For this reason, we call $\hat{\mathcal{M}}$ the *unbiased moment tensor*. It follows that an unbiased estimator of $\boldsymbol{A}$ is obtained by the optimization

$$\hat{J}[\boldsymbol{A}] = (\boldsymbol{A}; \hat{\mathcal{M}}\boldsymbol{A}) \rightarrow \min. \tag{11}$$

However, Eq. (8) involves the noise level ϵ, which is unknown. Here, we introduce an iterative scheme called *renormalization* [5], which adaptively removes the statistical bias without using the knowledge of the noise level ϵ and at the same time updates the weight $W_\alpha^{(kl)}$. The procedure is as follows:

1. Let $c = 0$ and $\boldsymbol{W}_\alpha = \boldsymbol{I}$, $\alpha = 1, ..., N$.
2. Compute the tensors $\mathcal{M}$, $\mathcal{N}^{(1)}$, and $\mathcal{N}^{(2)}$ given by Eqs. (8), (9), and (10), respectively.
3. Compute the smallest eigenvalue λ of the tensor
$$\hat{\mathcal{M}} = \mathcal{M} - c\mathcal{N}^{(1)} + c^2\mathcal{N}^{(2)}, \tag{12}$$
and the corresponding eigenmatrix $\boldsymbol{A}$ of unit norm.
4. If $\lambda \approx 0$, return $\boldsymbol{A}$. Else, update c and $\boldsymbol{W}_\alpha$ as follows:
$$D = \Big((\boldsymbol{A}; \mathcal{N}^{(1)}\boldsymbol{A}) - 2c(\boldsymbol{A}; \mathcal{N}^{(2)}\boldsymbol{A})\Big)^2 - 4\lambda(\boldsymbol{A}; \mathcal{N}^{(2)}\boldsymbol{A}), \tag{13}$$
$$c \leftarrow c + \frac{(\boldsymbol{A}; \mathcal{N}^{(1)}\boldsymbol{A}) - 2c(\boldsymbol{A}; \mathcal{N}^{(2)}\boldsymbol{A}) - \sqrt{D}}{2(\boldsymbol{A}; \mathcal{N}^{(2)}\boldsymbol{A})}, \tag{14}$$
$$\boldsymbol{W}_\alpha \leftarrow \Big(\boldsymbol{x}'_\alpha \times \boldsymbol{A}V_0[\boldsymbol{x}_\alpha]\boldsymbol{A}^\top \times \boldsymbol{x}'_\alpha + (\boldsymbol{A}\boldsymbol{x}_\alpha) \times V_0[\boldsymbol{x}'_\alpha] \times (\boldsymbol{A}\boldsymbol{x}_\alpha) + c[\boldsymbol{A}V_0[\boldsymbol{x}_\alpha]\boldsymbol{A}^\top \times V_0[\boldsymbol{x}'_\alpha]]\Big)_2^-. \tag{15}$$
If $D < 0$, Eq. (14) is replaced by $c \leftarrow c + \lambda/(\boldsymbol{A}; \mathcal{N}^{(1)}\boldsymbol{A})$.
5. Go back to Step 2.

6. Surface and Motion Parameters

After the transformation matrix $\boldsymbol{A}$ is obtained, we need to solve Problem 2 for computing the surface and motion parameters $\{\boldsymbol{n}, d\}$ and $\{\boldsymbol{h}, \boldsymbol{R}\}$. Since the scale of the transformation matrix $\boldsymbol{A}$ is indeterminate, we rescale it so that $|\boldsymbol{A}| = 1$. Also, we distinguish the following two types of camera motion:

Case 1: The camera moves on only one side of the planar surface; the two images are views from the same side of the surface.

Case 2: The camera penetrates through the planar surface; the second image is a view from the opposite side of the surface.

Then, the surface and motion parameters are computed by the following procedure [3], [7], [9]:

1. Let $\lambda_1 \geq \lambda_2 \geq \lambda_3$ (> 0) be the eigenvalues of matrix $\boldsymbol{A}^\top \boldsymbol{A}$, and $\{\boldsymbol{u}_1, \boldsymbol{u}_2, \boldsymbol{u}_3\}$ the orthonormal set of the corresponding eigenvectors. Let $\sigma_i = \sqrt{\lambda_i}$ (> 0), $i = 1, 2, 3$.

2. The surface parameters $\{\boldsymbol{n}, d\}$ in Case 1 are given by

$$\boldsymbol{n} = N[\sqrt{\sigma_1^2 - \sigma_2^2}\boldsymbol{u}_1 \pm \sqrt{\sigma_2^2 - \sigma_3^2}\boldsymbol{u}_3], \tag{16}$$

$$d = \frac{\sigma_2}{\sigma_1 - \sigma_3}, \tag{17}$$

   where the symbol $N[\cdots]$ denotes normalization to a unit vector. In Case 2, the above distance d is replaced by $d = \sigma_2/(\sigma_1 + \sigma_3)$.

3. The motion parameters $\{\boldsymbol{h}, \boldsymbol{R}\}$ in Case 1 are given by

$$\boldsymbol{h} = N[-\sigma_3\sqrt{\sigma_1^2 - \sigma_2^2}\boldsymbol{u}_1 \pm \sigma_1\sqrt{\sigma_2^2 - \sigma_3^2}\boldsymbol{u}_3], \tag{18}$$

$$\boldsymbol{R} = \frac{1}{\sigma_2}\left(\boldsymbol{I} + \sigma_2^3 \boldsymbol{p}\boldsymbol{h}^\top\right)\boldsymbol{A}^\top, \tag{19}$$

   and in Case 2 by

$$\boldsymbol{h} = N[\sigma_3\sqrt{\sigma_1^2 - \sigma_2^2}\boldsymbol{u}_1 \pm \sigma_1\sqrt{\sigma_2^2 - \sigma_3^2}\boldsymbol{u}_3], \tag{20}$$

$$\boldsymbol{R} = \frac{1}{\sigma_2}\left(-\boldsymbol{I} + \sigma_2^3 \boldsymbol{p}\boldsymbol{h}^\top\right)\boldsymbol{A}^\top, \tag{21}$$

   where the double sign $\pm$ corresponds to that in Eq. (16).

Thus, *eight* solutions are obtained. This ambiguity is due to the following two facts:

- The surface and motion parameters are computed from the transformation matrix $\boldsymbol{A}$, *not from individual feature points*.
- It is implicitly assumed that the camera can observe a surface *behind* the camera.

Suppose there exists no image noise. If the surface and motion parameters are $\{\boldsymbol{n}, d\}$ and $\{\boldsymbol{h}, \boldsymbol{R}\}$, respectively, the 3-D position of the αth feature point with position vectors $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ is $Z_\alpha \boldsymbol{x}_\alpha$ with respect to the first camera system and $Z'_\alpha \boldsymbol{x}'_\alpha$ with respect to the second camera system, where the *depths* Z_α and Z'_α are given by

$$Z_\alpha = \frac{d}{(\boldsymbol{n}, \boldsymbol{x}_\alpha)}, \quad Z'_\alpha = \frac{d - (\boldsymbol{n}, \boldsymbol{h})}{(\boldsymbol{n}, \boldsymbol{R}\boldsymbol{x}'_\alpha)}. \tag{22}$$

Hence, if the feature points are all *in front of the camera* before and after the camera motion, we can impose the condition $Z_\alpha > 0$ and $Z'_\alpha > 0$ for $\alpha = 1, ..., N$. Then, the number of the solution reduces to at most *two* and in most cases determined uniquely [4].

7. Optimal Correction of Feature Points

If the correct surface and motion parameters are chosen, the depths of each feature point is determined by Eqs. (22) if there exists no image noise. In the presence of image noise, however, the lines of sight determined by $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ do not meet on the surface determined by $\{\boldsymbol{n}, d\}$ (Fig. 2), which is equivalent to saying that Eq. (1) is not satisfied. Hence, we correct $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}'_\alpha$ into $\hat{\boldsymbol{x}}_\alpha$ and $\hat{\boldsymbol{x}}'_\alpha$ so that Eq. (1) is exactly satisfied. Letting $\hat{\boldsymbol{x}}_\alpha = \boldsymbol{x}_\alpha + \Delta\boldsymbol{x}_\alpha$ and $\hat{\boldsymbol{x}}'_\alpha = \boldsymbol{x}'_\alpha + \Delta\boldsymbol{x}'_\alpha$, substituting them into Eq. (1), and taking a first order approximation, we obtain

$$\Delta\boldsymbol{x}'_\alpha \times \boldsymbol{A}\boldsymbol{x}_\alpha + \boldsymbol{x}'_\alpha \times \Delta\boldsymbol{x}_\alpha = -\boldsymbol{x}'_\alpha \times \boldsymbol{A}\boldsymbol{x}_\alpha. \tag{23}$$

The correction is done optimally by minimizing the square sum of the *Mahalanobis distances*

$$J = (\Delta\boldsymbol{x}_\alpha, V_0[\boldsymbol{x}_\alpha]^- \Delta\boldsymbol{x}_\alpha) + (\Delta\boldsymbol{x}'_\alpha, V_0[\boldsymbol{x}'_\alpha]^- \Delta\boldsymbol{x}'_\alpha). \tag{24}$$

The solution is obtained in the following form:

$$\Delta\boldsymbol{x}_\alpha = -(V_0[\boldsymbol{x}_\alpha]\boldsymbol{A}^\top \times \boldsymbol{x}'_\alpha)\boldsymbol{W}(\boldsymbol{x}'_\alpha \times \boldsymbol{A}\boldsymbol{x}_\alpha),$$
$$\Delta\boldsymbol{x}'_\alpha = (V_0[\boldsymbol{x}'_\alpha] \times (\boldsymbol{A}\boldsymbol{x}_\alpha))\boldsymbol{W}(\boldsymbol{x}'_\alpha \times \boldsymbol{A}\boldsymbol{x}_\alpha), \tag{25}$$

$$\boldsymbol{W} = \Big(\boldsymbol{x}'_\alpha \times \boldsymbol{A}V_0[\boldsymbol{x}_\alpha]\boldsymbol{A}^\top \times \boldsymbol{x}'_\alpha + (\boldsymbol{A}\boldsymbol{x}_\alpha) \times V_0[\boldsymbol{x}'_\alpha] \times (\boldsymbol{A}\boldsymbol{x}_\alpha)\Big)_2^-. \tag{26}$$

For a vector $\boldsymbol{a} = (a_i)$ and a matrix $\boldsymbol{U} = (U_{ij})$, we write $\boldsymbol{a} \times \boldsymbol{U}$ and $\boldsymbol{U} \times \boldsymbol{a}$ to denote the matrices whose (i, j) elements are $\sum_{k,l=1}^3 \epsilon_{ikl} a_k U_{lj}$ and $\sum_{k,l=1}^3 \epsilon_{jkl} U_{ik} a_l$, respectively. Since Eq. (24) is a first order approximation, the corrected values $\hat{\boldsymbol{x}}_\alpha$ and $\hat{\boldsymbol{x}}'_\alpha$ may not exactly satisfy Eq. (1). So, the above correction is iterated until Eq. (1) is sufficiently satisfied. From the resulting $\hat{\boldsymbol{x}}_\alpha$ and $\hat{\boldsymbol{x}}'_\alpha$, the depths Z_α and Z'_α are computed by using Eqs. (22).

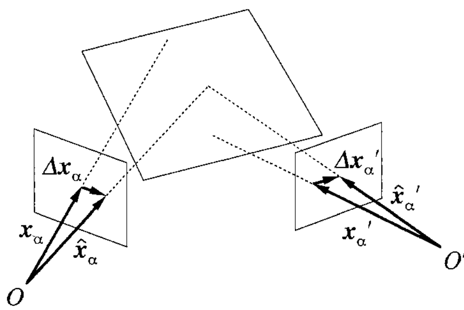


Fig. 2 Optimal correction.

8. Numerical Simulation

Figure 3 shows two simulated motion images (512×512 pixels) of a planar grid. The focal length is assumed to be $f = 600$ (pixels). Gaussian noise with standard deviation of five pixels is added to the x and y coordinates of each grid point independently, and the surface and motion parameters $\{\boldsymbol{n}, d\}$ and $\{\boldsymbol{h}, \boldsymbol{R}\}$ are computed. The deviation from their true values $\{\bar{\boldsymbol{n}}, \bar{d}\}$ and $\{\bar{\boldsymbol{h}}, \bar{\boldsymbol{R}}\}$ is measured as follows:

- The error in the surface parameters is represented by

$$\Delta \boldsymbol{u} = \boldsymbol{P}_{\bar{\boldsymbol{n}}}(\boldsymbol{n} - \bar{\boldsymbol{n}}) + \frac{\|\bar{\boldsymbol{h}}\| d - \bar{d}}{\bar{d}} \bar{\boldsymbol{n}}, \tag{27}$$

where $\boldsymbol{P}_{\bar{\boldsymbol{n}}} = \boldsymbol{I} - \bar{\boldsymbol{n}}\bar{\boldsymbol{n}}^\top$ is the projection matrix onto the plane orthogonal to $\bar{\boldsymbol{n}}$.

- The error in translation is represented by

$$\Delta \boldsymbol{h} = \boldsymbol{P}_{N[\bar{\boldsymbol{h}}]}(\boldsymbol{h} - N[\bar{\boldsymbol{h}}]), \tag{28}$$

where $\boldsymbol{P}_{N[\bar{\boldsymbol{h}}]} = \boldsymbol{I} - N[\bar{\boldsymbol{h}}]N[\bar{\boldsymbol{h}}]^\top$ is the projection matrix onto the plane orthogonal to $\bar{\boldsymbol{h}}$.

- The error in rotation is represented by

$$\Delta \boldsymbol{\Omega} = \Delta \tilde{\Omega} \tilde{l}, \tag{29}$$

where $\tilde{\Omega}$ and $\tilde{l}$ are, respectively, the axis and angle of the relative rotation $\tilde{\boldsymbol{R}} = \boldsymbol{R}\bar{\boldsymbol{R}}^{-1}$ $(= \boldsymbol{R}\bar{\boldsymbol{R}}^\top)$.

Figures 4(a), 5(a), and 6(a) plot $\Delta \boldsymbol{u}$, $\Delta \boldsymbol{h}$, and $\Delta \boldsymbol{\Omega}$, respectively, in three dimensions for 100 trials, each time using different noise. As a comparison, Figs. 4(b), 5(b), and 6(b) show the corresponding result computed by the optimal least-squares approximation (the weights $W_\alpha^{(kl)}$ are computed from the true values). It is clearly seen that statistical bias exists for the least-squares approximation and the bias is removed by renormalization.

Figure 7(a) shows one example of a reconstructed grid. This example corresponds to the white dots in Figs. 4–6. The true position is drawn in dashed lines. Figure 7(b) shows the reconstructed surface from the same data without applying the optimal correction. We can see that the correction enhances the accuracy of 3-D

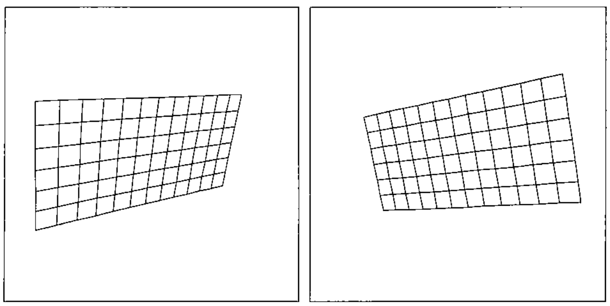
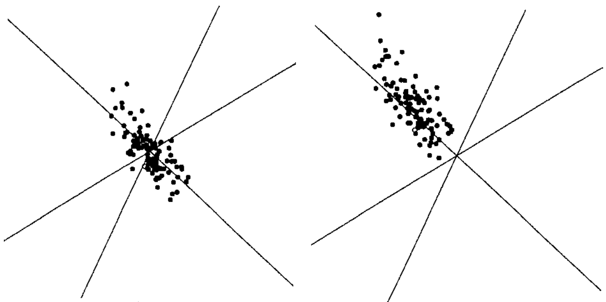
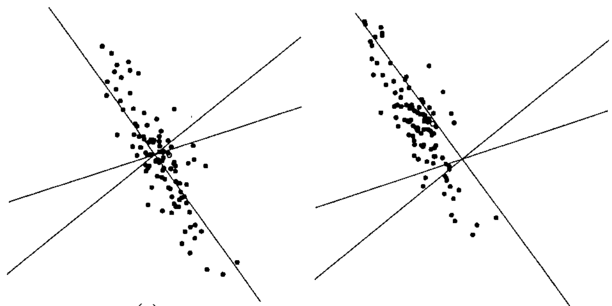


Fig. 3 Motion images of a planar grid.



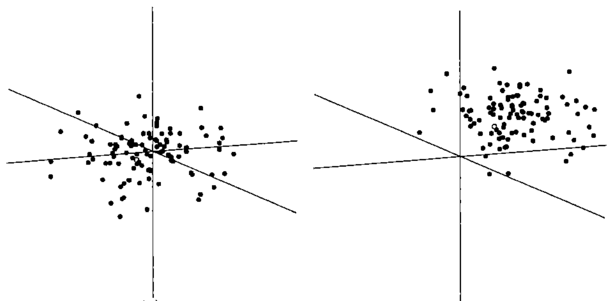
(a) (b)

Fig. 4 Errors in the surface parameters. (a) Optimal solution. (b) Least-squares solution.



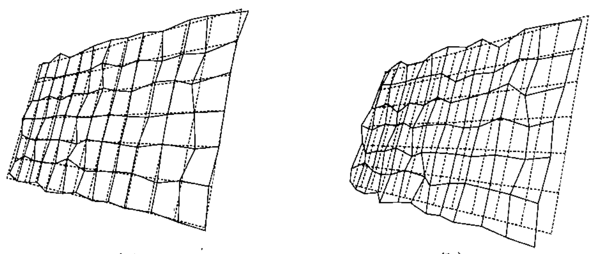
(a) (b)

Fig. 5 Errors in translation. (a) Optimal solution. (b) Least-squares solution.



(a) (b)

Fig. 6 Errors in rotation. (a) Optimal solution. (b) Least-squares solution.



(a) (b)

Fig. 7 (a) Optimal 3-D reconstruction. (b) Simple 3-D reconstruction.

reconstruction.

9. Concluding Remarks

In this paper, we have presented a *theoretically best* algorithm within the framework of our image noise model for reconstructing 3-D from two views for coplanar feature points. Pointing out that statistical bias is introduced if the least-squares scheme is used in the presence of image noise, we have proposed a scheme called *renormalization*, which automatically removes statistical bias. We have also presented an optimal correction scheme for canceling the effect of image noise in individual feature points. Finally, we showed numerical simulation and confirmed the effectiveness of our method.

One remaining issue is how to switch from the general algorithm to the planar surface algorithm when we have no knowledge about the scene. In the past, many researchers endorsed *using the general algorithm first* and switching to the planar surface algorithm *when the general algorithm fails* due to such anomalies as multiplicities of eigenvalues and zero division. However, we endorse *using the planar surface algorithm first* and switching to the general algorithm *when the solution does not sufficiently fit to the observed image motion* for an estimated noise level. To be specific, let ϵ be the noise level expected from the accuracy of image processing, and $\hat{\epsilon}$ its *a posteriori estimate* computed by

$$\hat{\epsilon}^2 = \frac{c}{1 - 4/N}, \tag{30}$$

where N is the number of the feature points and c is the value returned by the renormalization procedure. It can be shown that $2(N-4)\hat{\epsilon}^2/\epsilon^2$ is a χ^2 random variable with $2(N-4)$ degrees of freedom. Hence, the hypothesis that the object is planar is rejected with significance level $a\%$ if

$$\frac{\hat{\epsilon}^2}{\epsilon^2} > \frac{\chi^2_{2(N-4),a}}{2(N-4)}, \tag{31}$$

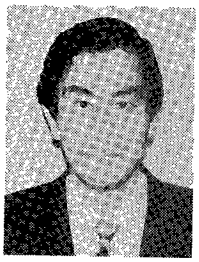
where $\chi^2_{2(N-4),a}$ is the upper $a\%$ point of the χ^2 distribution with $2(N-4)$ degrees of freedom.

Acknowledgments

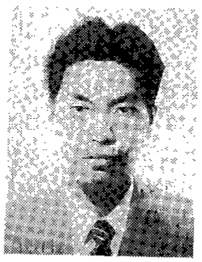
This work was in part supported by the Ministry of Education, Science, Sports, and Culture, Japan under a Grant in Aid for Scientific Research B (No. 07458067) and the Okawa Institute of Information and Telecommunication, Japan.

References

[1] Faugeras, O., *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, MA, U.S.A., 1993.
[2] Kanatani, K., "Unbiased estimation and statistical analysis of 3-D rigid motion from two views," *IEEE Trans. Patt. Anal. Mach. Intell.* vol.15, no.1, pp.37–50, Jan. 1993.
[3] Kanatani, K., *Geometric Computation for Machine Vision*, Oxford Univ. Press, Oxford, 1993.
[4] Kanatani, K., *Statistical Optimization for Geometric Computation: Theory and Practice*, Lecture Note, Department of Computer Science, Gunma University, 2nd Edition, Jun. 1995.
[5] Kanatani, K., "Statistical analysis of geometric computation," *CVGIP: Image Understanding*, vol.59, no.3, pp.286–306, 1994.
[6] Kanatani, K., "Renormalization for motion analysis: Statistically optimal algorithm," *IEICE Trans. Inf. & Syst.*, vol.E77-D, no.11, pp.1233–1239, Nov. 1994.
[7] Longuet-Higgins, H.C., "The reconstruction of a plane surface from two perspective projections," *Proc. Roy. Soc. Lond.*, ser.B, vol.227, no.1249, pp.399–410, 1986.
[8] Maybank, S., *Theory of Reconstruction from Image Motion*, Springer, Berlin, 1993.
[9] Weng, J., Huang, T.S. and Ahuja, N., *Motion and Structure from Image Sequences*, Springer, Berlin, 1993.



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 Univ. Press, 1993).



Sachio Takeda received his Ms. in computer science from Gunma University in 1995. He is currently at NTT.