PAPER Special Issue on Computer Vision and Its Applications

Anatomy of Camera Calibration Using Vanishing Points

Kenichi KANATANI†, *Member and* Yasuhiro ONODERA††, *Nonmember*

SUMMARY A new mathematical formalism is proposed for constructing elements of camera calibration that measure the focal length and the orientation of the camera: the focal length and the camera orientation are computed by detecting, on the image plane, the vanishing points of two sets of lines that are mutually orthogonal in the scene; the distance of the scene coordinate origin from the camera is determined by locating, on the image plane, a point whose scene coordinates are known. We show that the separation of the calibration process into atomic modules enables us to not only predict theoretically optimal estimates but also estimate their reliability.

1. Introduction

Visual control using video cameras is one of the most essential components for intelligent robot operations. To this end, the study of computer vision is expected to play a central role. However, almost all techniques for extracting 3-D information are based on the assumption that the camera imaging geometry is known. Whenever we try to implement any computer vision technique by using a real camera, we immediately face the difficulty of accurately calibrating the camera. Today, more and more people agree that the difficulty of camera calibration is one of the major obstacles that prevent the use of computer vision techniques in real environments[(1)-(5),(9)-(13)].

In computer vision studies, the camera imaging is usually modeled as perspective projection from the origin O (called the "viewpoint") of the camera-based XYZ-coordinate system onto an image plane placed parallel to the XY-plane in distance f, which is often referred to as the "focal length", from the viewpoint O (Fig. 1). Let us call the parameters that specify the 3-D position of the viewpoint O and the 3-D orientation of the camera XYZ-coordinate system the "pose parameters".

The camera calibration techniques reported in the past involve very complicated procedures. One reason is that these techniques aim to take all factors into consideration, including the optical distortion (called "abberation"), the distortion of the mapping from the camera to the display by raster scanning, and the digitization into discrete pixels. Essentially, most of the reported techniques take the following "parametric fitting" approach:

1. Set, in the scene, multiple reference points whose 3-D coordinates are known.
2. Locate the images of the reference points on the video display.
3. Construct a parameterized camera imaging model by taking into account all conceivable factors—perspective projection, lens distortion, raster scanning, discrete pixels, etc.
4. Express the 2-D image coordinates of the reference points in terms of the model parameters, assuming that the imaging model is correct.
5. Determine the parameters by minimizing, say in the sense of least-squares, the discrepancy between the image coordinates of the observed reference points and their predicted locations.

We oppose this approach on the following ground.

- The quantity to be minimized is a complicated nonlinear function of the calibration parameters, and analytic solutions are difficult to obtain. So, we must resort to numerical search by iterations. However, it is in general not easy to guarantee convergence for such iterations.
- If the camera imaging model is a pure perspective projection, the model equations can be made linear in appearance by introducing artificial variables (e.g., 3-D homogeneous coordinates). Hence, it appears that a simple least-squares method

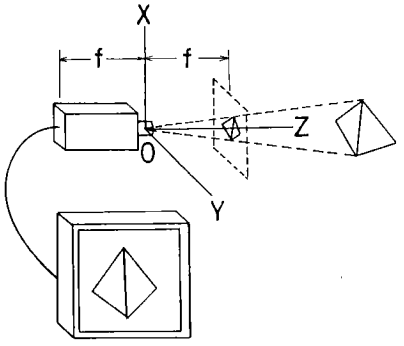

Fig. 1 Camera imaging geometry.

Manuscript received February 14, 1991.
Manuscript revised May 9, 1991.

† The author is with the Department of Computer Science, Gunma University, Kiryu-shi, 376 Japan.
†† The author is with Atsugi Technology Center, Sony Corporation, Atsugi-shi, 243 Japan.

can be applied. However, the least-squares method makes sense only when error behaviors are well understood. It is extremely dangerous to treat noisy variables and artificial variables that are immune to noise in the same way without knowing their geometric meanings and error behaviors.

- Even if the computed parameters, as a whole, attain a minimum, this does not mean that each of the parameters is reliable. Suppose, for example, quantity J is to be minimized but it is not very sensitive to one parameter, say α, as compared with another parameter, say β, near the optimum: $|\partial J / \partial \alpha| \ll |\partial J / \partial \beta|$. Then, the estimated value of α may be largely distorted to compensate for the error in β. Thus, this approach becomes extremely dangerous as the number of the parameters increases, and especially so when parameters of different geometric origins such as the focal length and the center of the image are mixed together.
- The mechanism of estimation is different from parameter to parameter. For example, the focal length cannot be detected accurately from images unless the effect of "foreshortening" is strong, because assuming different focal lengths does not affect the resulting 3-D interpretation very much if foreshortening is not apparent. This means that the computed focal length becomes more reliable as the effect of foreshortening becomes stronger. Hence, we must first check the "estimation mechanism" for each parameter, and then arrange the setup accordingly so that the reliability is maximized. This kind of consideration is impossible if all the parameters are optimized as a single step.
- All the calibration parameters are not equally important. These days, for example, lens distortion is small thanks to the advanced manufacturing technology. Hence, if we use a well manufactured high quality camera, we need not worry very much about abberation, while other parameters, e.g., the focal length, may be vital in some applications. Thus, it is desirable that the parameters we are interested in can be estimated separately.

In view of these observations, we propose to separate the calibration process into "atomic modules", each based on a simple and well understood geometric relationship. In this paper, we focus on the focal length estimation module and the pose parameter estimation module, since the focal length and the pose parameters are frequently changed, and hence must be recalibrated quickly.

In this paper, we consider a scheme of using a specially designed calibration board, which plays the role of the scene coordinate system. It would be desirable to use non-coplanar reference points from a theoretical point of view, but in practice it is not easy to handle non-coplanar reference points efficiently. Instead of reference points, we consider the use of two sets of mutually orthogonal parallel lines, because lines are expected to be more robust than points. Estimation of the focal length is based on their vanishing points; the distance of the scene coordinate origin from the camera is determined by locating a point whose position on the calibration board is known.

In our formulation, all points and lines are represented by unit vectors, which we call "N-vectors"[6],[7]. In terms of N-vectors, points at infinity and the line at infinity can be treated as if they are ordinary points and a line. This formulation enables us to incorporate a statistical consideration of error behaviors, computing not only optimal estimates but also the variance of each estimate, from which we can deduce a quantitative "confidence level" of the estimate.

This paper does not discuss the overall system organization of camera calibration and its performance, since they are affected by many factors including the geometric correction of lens abberation, which should be treated independently of other camera parameters.

2. The Pose Parameters of the Camera

Take an XYZ camera coordinate system with origin O (the viewpoint), and fix an $\bar{X}\bar{Y}\bar{Z}$ scene coordinate system in the scene with origin $\bar{O}$ (Fig. 2). Let $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, and $\boldsymbol{e}_3$ be the unit vectors along the $\bar{X}$-, $\bar{Y}$-, and $\bar{Z}$-axes, respectively. Let $\boldsymbol{m}_o$ be the unit vector starting from the viewpoint O and poiniting toward the scene coordinate origin $\bar{O}$. We call $\boldsymbol{m}_o$ the "N-vector" of $\bar{O}$. Let $r_o = |O\bar{O}|$ be the distance of the scene coordinate origin $\bar{O}$ from the viewpoint O.

We regard the XYZ camera coordinate system as obtained by (1) first rotating the $\bar{X}\bar{Y}\bar{Z}$ scene coordinate system around its origin $\bar{O}$ by a rotation matrix $\boldsymbol{R}$ and (2) then translating it by a vector $\boldsymbol{h}$, where the components of $\boldsymbol{R}$ and $\boldsymbol{h}$ are defined with respect to the $\bar{X}\bar{Y}\bar{Z}$-coordinate system. Let us call $\{\boldsymbol{R}, \boldsymbol{h}\}$ the "pose parameters".

Let $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ be the unit vectors along the X-, Y-, and Z-axes, respectively. The above definition of

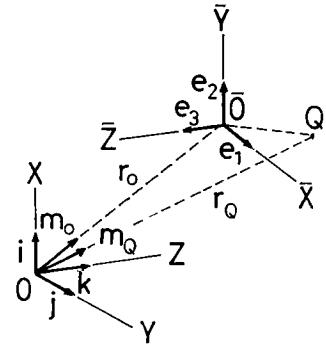

Fig. 2 The camera coordinate system and the scene coordinate system.

$\boldsymbol{R} = (\boldsymbol{R}_{ij})$ and $\boldsymbol{h}=(h_i)$ is rephrased as

$$\begin{aligned} \boldsymbol{i} &= R_{11}\,\boldsymbol{e}_1 + R_{21}\,\boldsymbol{e}_2 + R_{31}\,\boldsymbol{e}_3, \\ \boldsymbol{j} &= R_{12}\,\boldsymbol{e}_1 + R_{22}\,\boldsymbol{e}_2 + R_{32}\,\boldsymbol{e}_3, \qquad (1) \\ \boldsymbol{k} &= R_{13}\,\boldsymbol{e}_1 + R_{23}\,\boldsymbol{e}_2 + R_{33}\,\boldsymbol{e}_3, \end{aligned}$$

$$\overrightarrow{OO} = h_1\,\boldsymbol{e}_1 + h_2\,\boldsymbol{e}_2 + h_3\,\boldsymbol{e}_3. \qquad (2)$$

Then, the pose parameters are computed as follows (the proof is easy). Let

$$\boldsymbol{e}_1 = \begin{bmatrix} e_{1(1)} \\ e_{1(2)} \\ e_{1(3)} \end{bmatrix}, \quad \boldsymbol{e}_2 = \begin{bmatrix} e_{2(1)} \\ e_{2(2)} \\ e_{2(3)} \end{bmatrix}, \quad \boldsymbol{e}_3 = \begin{bmatrix} e_{3(1)} \\ e_{3(2)} \\ e_{3(3)} \end{bmatrix},$$

$$\boldsymbol{m}_o = \begin{bmatrix} m_{o(1)} \\ m_{o(2)} \\ m_{o(3)} \end{bmatrix} \qquad (3)$$

be the components of vectors $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, $\boldsymbol{e}_3$, and $\boldsymbol{m}_o$ expressed with respect to the XYZ camera coordinate system. The pose parameters $\{\boldsymbol{R}, \boldsymbol{h}\}$ are given by

$$\boldsymbol{R} = \begin{bmatrix} e_{1(1)} & e_{1(2)} & e_{1(3)} \\ e_{2(1)} & e_{2(2)} & e_{2(3)} \\ e_{3(1)} & e_{3(2)} & e_{3(3)} \end{bmatrix}, \qquad (4)$$

$$\boldsymbol{h} = -\,r_o \begin{bmatrix} R_{11}m_{o(1)} + R_{12}m_{o(2)} + R_{13}m_{o(3)} \\ R_{21}m_{o(1)} + R_{22}m_{o(2)} + R_{23}m_{o(3)} \\ R_{31}m_{o(1)} + R_{32}m_{o(2)} + R_{33}m_{o(3)} \end{bmatrix}. \qquad (5)$$

3. Determination of the Motion Parameters

Suppose the camera is moved in the scene. In order to specify the position and orientation of the $X'Y'Z'$-coordinate system after the motion relative to the XYZ-coordinate system before the motion, we regard the $X'Y'Z'$-coordinate system as obtained by (1) first rotating the XYZ-coordinate system around its origin O by a rotation matrix $\tilde{\boldsymbol{R}}$ and (2) then translating it by a vector $\tilde{\boldsymbol{h}}$, where the components of $\tilde{\boldsymbol{R}}$ and $\tilde{\boldsymbol{h}}$ are defined with respect to the XYZ-coordinate system. Let us call $\{\tilde{\boldsymbol{R}}, \tilde{\boldsymbol{h}}\}$ the "motion parameters" (Fig. 3).

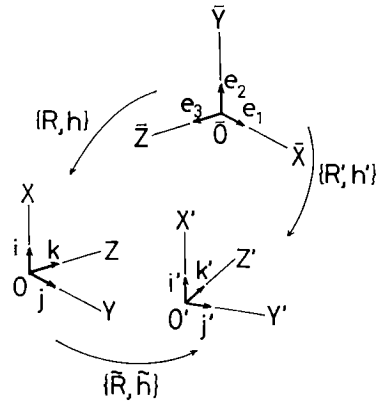

Fig. 3 Motion parameters.

Let $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k}$ be, respectively, the unit vectors along the original X-, Y-, and Z-axes, and let $\boldsymbol{i}'$, $\boldsymbol{j}'$, and $\boldsymbol{k}'$ be, respectively, the unit vectors along the X'-, Y'-, and Z'-axes after the motion. The above definition of $\tilde{\boldsymbol{R}} = (\tilde{R}_{ij})$ and $\boldsymbol{h} = (\tilde{h}_i)$ is rephrased as

$$\begin{aligned} \boldsymbol{i}' &= \tilde{R}_{11}\,\boldsymbol{i} + \tilde{R}_{21}\,\boldsymbol{j} + \tilde{R}_{31}\,\boldsymbol{k}, \\ \boldsymbol{j}' &= \tilde{R}_{12}\,\boldsymbol{i} + \tilde{R}_{22}\,\boldsymbol{j} + \tilde{R}_{32}\,\boldsymbol{k}, \qquad (6) \\ \boldsymbol{k}' &= \tilde{R}_{31}\,\boldsymbol{i} + \tilde{R}_{23}\,\boldsymbol{j} + \tilde{R}_{33}\,\boldsymbol{k}, \end{aligned}$$

$$\overrightarrow{OO'} = \tilde{h}_1\,\boldsymbol{i} + \tilde{h}_2\,\boldsymbol{j} + \tilde{h}_3\,\boldsymbol{k}. \qquad (7)$$

Then, the motion parameters are computed as follows (the proof is easy). If the pose parameters of the camera are $\{\boldsymbol{R}, \boldsymbol{h}\}$ and $\{\boldsymbol{R}', \boldsymbol{h}'\}$ before and after the motion, respectively, the motion parameters $\{\tilde{\boldsymbol{R}}, \tilde{\boldsymbol{h}}\}$ are given by

$$\tilde{\boldsymbol{R}} = \boldsymbol{R}^T\boldsymbol{R}', \quad \tilde{\boldsymbol{h}} = \boldsymbol{R}^T(\boldsymbol{h}' - \boldsymbol{h}). \qquad (8)$$

4. Determination of the Absolute Depth

In order to determine the distance r_o of the scene coordinate origin $\bar{O}$ from the viewpoint O, we need some information about absolute length in the scene. Here, we observe, on the $\bar{X}\bar{Y}$-plane, a fixed a point Q whose distance $|\,\bar{O}Q\,|$ from $\bar{O}$ is known (Fig. 2). Let $\boldsymbol{m}_Q$ be the unit vector starting from the viewpoint O and pointing toward Q; we call $\boldsymbol{m}_Q$ the "N-vector" of point Q. Let $(\boldsymbol{a}, \boldsymbol{b})$ designate the inner product of vectors $\boldsymbol{a}$ and $\boldsymbol{b}$, and $\|\,\boldsymbol{a}\,\|$ the norm of vector $\boldsymbol{a}$. It is easy to confirm that the distance r_o of the scene coordinate origin $\bar{O}$ from the viewpoint O is given by

$$r_o = \frac{|\,(\boldsymbol{m}_Q, \boldsymbol{e}_3)\,| \cdot |\,\bar{O}Q\,|}{\|\,(\boldsymbol{m}_o, \boldsymbol{e}_3)\,\boldsymbol{m}_Q - (\boldsymbol{m}_Q, \boldsymbol{e}_3)\,\boldsymbol{m}_o\,\|}. \qquad (9)$$

5. N-vectors of Points and Lines

Given a point P in the scene, we call the unit vector $\boldsymbol{m}$ starting from the viewpoint O and pointing toward P the "N-vector"[7] of point P. If we use the

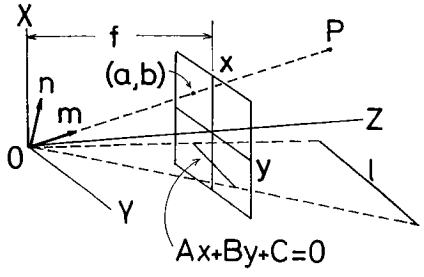

Fig. 4 The N-vectors of a point and a line.

notation $N[\boldsymbol{a}] = \boldsymbol{a}/\|\boldsymbol{a}\|$ to designate the normalization of vector $\boldsymbol{a}$ into a unit vector, the N-vector $\boldsymbol{m}$ of a point P whose image coordinates are (a, b) is given by

$$\boldsymbol{m} = N\begin{bmatrix} a \\ b \\ f \end{bmatrix}. \tag{10}$$

Given a line l in the scene, we define its "N-vector"[7] as the unit vector $\boldsymbol{n}$ normal to the plane passing through the viewpoint O and intersecting the image plane along l (Fig. 4). It is easy to see that if the projection image of l is $Ax + By + C = 0$, its N-vector $\boldsymbol{n}$ is given by

$$\boldsymbol{n} = \pm N\begin{bmatrix} A \\ B \\ C/f \end{bmatrix}, \tag{11}$$

where the sign is arbitrarily chosen.

From Fig. 4, we can see that a point of N-vector $\boldsymbol{m}$ is on a line of N-vector $\boldsymbol{n}$ if and only if $(\boldsymbol{m}, \boldsymbol{n}) = 0$. Hence, if point P is on both lines l_1 and l_2, N-vector $\boldsymbol{m}$ must be orthogonal to both $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$. Thus, the N-vector $\boldsymbol{m}$ of the intersection P of two lines l_1 and l_2 given by

$$\boldsymbol{m} = \pm N[\boldsymbol{n}_1 \times \boldsymbol{n}_2], \tag{12}$$

where $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ are the N-vectors of l_1 and l_2, respectively, and the sign is chosen so that the Z-component becomes nonnegative.

Lines that meet at a common intersection on the image plane are said to be "concurrent". Computation of the common intersection of concurrent lines plays an essential role in our calibration procedure. However, if lines are obtained from real data by image processing, error is inevitable; lines that are supposed to be concurrent may not concurrent. Hence, we need to estimate a common intersection of not necessarily concurrent lines (Fig. 5). The common intersection may not be found within the image frame; it may be located at infinity if the lines happen to be parallel on the image plane. In view of this, the computation should be done in terms of N-vectors.

Let $\boldsymbol{n}_1, \cdots, \boldsymbol{n}_N$ be the N-vectors of not necessarily concurrent lines. If these lines exactly pass through a point of N-vector $\boldsymbol{m}$, we have $(\boldsymbol{m}, \boldsymbol{n}_\alpha) = 0$, $\alpha = 1, \cdots, N$. Hence, $\boldsymbol{m}$ is estimated by minimizing $\sum_{\alpha=1}^{N} W_\alpha (\boldsymbol{m}, \boldsymbol{n}_\alpha)^2$ $(= (\boldsymbol{m}, (\sum_{\alpha=1}^{N} W_\alpha \boldsymbol{n}_\alpha \boldsymbol{n}_\alpha^T)\boldsymbol{m}))$ by appropriately introducing a weight w_α to each line. Since this is a quadratic form in $\boldsymbol{m}$, it is minimized by the unit eigenvector of the "moment matrix"

$$\boldsymbol{N} = \sum_{\alpha=1}^{N} W_\alpha \boldsymbol{n}_\alpha \boldsymbol{n}_\alpha^T \tag{13}$$

for the smallest eigenvalue[6],[7]. The sign is chosen so that the Z-component becomes nonnegative.

The weights W_α should be determined in such a way that the resulting estimate is most robust to noise. The optimal weights and the "covariance matrix" of the resulting optimal estimate are given as follows (the proof requires many mathematical preliminaries and long derivation):

[Proposition 1] The optimal weights W_α for estimating a common intersection are given by

$$W_\alpha = \frac{1}{(\boldsymbol{m}, V[\boldsymbol{n}_\alpha]\boldsymbol{m})}, \tag{14}$$

where $\boldsymbol{n}_\alpha$ is the N-vector of the αth line, $\boldsymbol{m}$ the N-vector of the intersection, and $V[\boldsymbol{n}_\alpha]$ the covariance matrix of $\boldsymbol{n}_\alpha$ (see Appendix A). □

[Proposition 2] The covariance matrix $V[\boldsymbol{m}]$ of the N-vector $\boldsymbol{m}$ of an optimally estimated common intersection is given by

$$V[\boldsymbol{m}] = \frac{\boldsymbol{u}\boldsymbol{u}^T}{\lambda_u} + \frac{\boldsymbol{v}\boldsymbol{v}^T}{\lambda_v}, \tag{15}$$

where $\boldsymbol{u}$ and $\boldsymbol{v}$ are the unit eigenvectors of the moment matrix $\boldsymbol{M}$ other than $\boldsymbol{m}$, and λ_u and λ_v their respective eigenvalues. □

The optimal weights W_α contain the N-vector $\boldsymbol{m}$ that we want to compute. Hence, in practice, we must use an appropriate estimate of $\boldsymbol{m}$, say, the value estimated with uniform weights. In order to apply the procedures described above, we need not know the true value of the focal length f; we can use an estimate of f for computing N-vectors. As long as the same f is used (including the optimization), the result is not significantly affected.

If the value of f is altered, it is clear from Eqs.

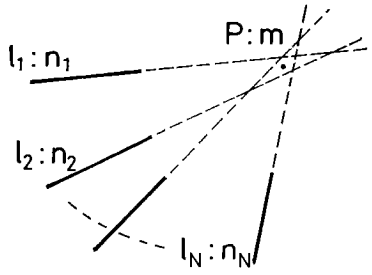

Fig. 5 Estimation of a common intersection.

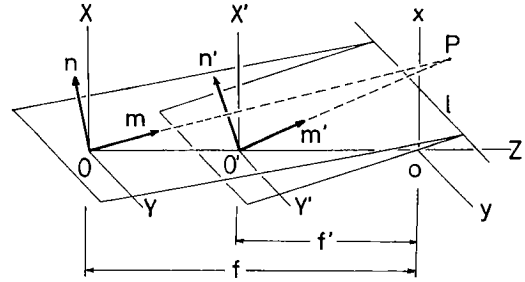

Fig. 6 The focal length f and N-vectors.

(10) and (11) and Fig. 6 that if $\boldsymbol{m} = (m_1, m_2, m_3)^T$ is the N-vector of a point and $\boldsymbol{n} = (n_1, n_2, n_3)^T$ is the N-vector of a line with respect to focal length f, the corresponding N-vectors $\boldsymbol{m}'$ and $\boldsymbol{n}'$ with respect to another focal length f' are respectively given by

$$\boldsymbol{m}' = N\left[\begin{bmatrix} m_1 \\ m_2 \\ (f'/f)\, m_3 \end{bmatrix}\right],$$

$$\boldsymbol{n}' = N\left[\begin{bmatrix} n_1 \\ n_2 \\ (f/f')\, n_3 \end{bmatrix}\right]. \tag{16}$$

6. Determination of the Focal Length

If an infinitely long line in the scene is projected onto the image plane, its image terminates at a point called the "vanishing point". From Fig. 7, it is easy to confirm that the N-vector of the vanishing point of a line in the scene indicates its 3-D orientation[6],[7].

Hence, if lines parallel in the scene are projected onto the image plane, they meet at their common vanishing point. From this fact, we obtain the following procedure[7],[9]:

1. Take an image of two mutually orthogonal sets of parallel lines in the scene.
2. Assuming a tentative value $\hat{f}$ for the focal length, compute the N-vectors $\boldsymbol{m} = (m_1, m_2, m_3)^T$ and $\boldsymbol{m}' = (m_1', m_2', m_3')^T$ of the vanishing points of these lines by the method described in Sect. 5.
3. From Eqs. (16), the true focal length f is given by

$$f = \hat{f}\sqrt{-\frac{m_1 m_1' + m_2 m_2'}{m_3 m_3'}}. \tag{17}$$

Equation (17) yields a less reliable value as the vanishing points move farther away from the image origin; both the numerator and the denominator in the square-root of Eq. (17) approach 0. This instability is inevitable, since the focal length affects the image in the form of "foreshortening". If the vanishing points are far apart from the image origin, the image is affected little (except for the scale) by the value of f, which means that the focal length f cannot be determined reliably. On the other hand, if the vanishing points are close to the image origin, some line segments may become very short, which decreases the reliability of line fitting, or some line segments may come very close to each other, which decreases the reliability of vanishing point detection.

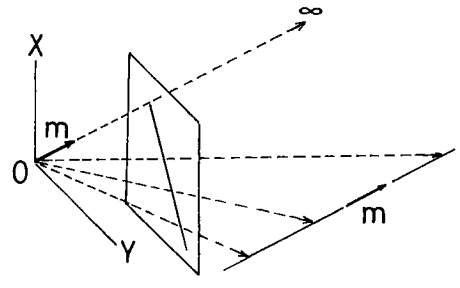

Fig. 7 The vanishing point of a line in the scene.

One way to increase the reliability is repeating the measurement in different settings and average the result. However, if we take the direct average, a small number of very deviated values (the so called "outliers") may damage the result. This is avoided if we take a weighted average in such a way that unreliable data are given small weights while reliable data are given large weights. To this end, we must compute the "variance" of f. For simplicity, assume that the current focal length $\hat{f}$ is a fairly accurate estimate of the true f. Here, we list the final result without a proof (a complete proof would require many mathematical preliminaries and long derivation procedures).

[Proposition 3] If $\boldsymbol{m}$ and $\boldsymbol{m}'$ are independent data, and if $\hat{f} = f$, the variance of f is given by

$$V[f] = \frac{f^2}{4}\frac{(\boldsymbol{m}', V[\boldsymbol{m}]\,\boldsymbol{m}') + (\boldsymbol{m}, V[\boldsymbol{m}']\boldsymbol{m}')}{(m_3 m_3')^2}, \tag{18}$$

where $V[\boldsymbol{m}]$ and $V[\boldsymbol{m}']$ are the "covariance matrices" (given by Propositiong 2) of $\boldsymbol{m}$ and $\boldsymbol{m}'$, respectively. □

If the vanishing points are far apart from the image orign, the denominator becomes small. If the vanishing points are close to the image origin, and some line segments are very short or very close to each other, the covariance matrices $V[\boldsymbol{m}]$ and $V[\boldsymbol{m}']$ grow large (cf. Appendix A and Proposition 2). Thus, the most reliable value is obtained when the variance $V[f]$ of Eq. (18) takes its minimum.

Suppose we repeat the measurement N times in different settings, and obtain values $\{f_\alpha\}$ with variances $\{V[f_\alpha]\}$. Let us take a weighted average in such a way that the resulting $\bar{f}$ is most reliable, which we interpret as having a minimum variance. Then, we obtain the following result (we omit the proof):

[Proposition 4] For independent N data $\{f_\alpha\}$, the optimal estimate is given by

$$\bar{f} = \sum_{\alpha=1}^{N} \frac{f_\alpha}{V[f_\alpha]} \Big/ \sum_{\alpha=1}^{N} \frac{1}{V[f_\alpha]}, \tag{19}$$

and its variance is given by

$$V[\bar{f}] = 1/\sum_{\alpha=1}^{N} \frac{1}{V[f_\alpha]}, \tag{20}$$

where $V[f_\alpha]$ is the variance (given by Eq. (18)) of f_α.

Thus, we can obtain not only the optimal estimate $\bar{f}$ but also its variance $V[\bar{f}]$, from which we can deduce the reliability of the $\bar{f}$. For example, we can conclude that the true focal length f is in the range $\bar{f}$

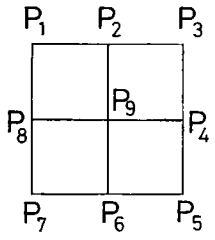

Fig. 8 The square grid pattern of the calibration board.

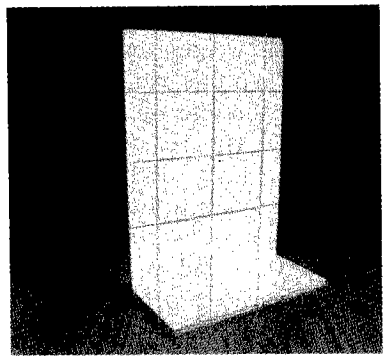

Fig. 9 Line fitting of the calibration board image.

$\pm 1.96\sqrt{V[\bar{f}]}$ with 95% confidence, assuming that the error distribution is approximately Gaussian.

7. Procedure of Focal Length Calibration

The actual procedure to determine the focal length is as follows. We use a calibration board on which the square grid pattern of Fig. 8 is drawn.

[Calibration of the focal length]

1. Detect line segments in the calibration pattern image, say, by the Hough transform or by manual specification though an interactive interface.
2. Fit lines to the detected line segments (Fig. 9) by the least-squares method, and determine the N-vectors of the fitted lines with respect to an estimate $\hat{f}$ of the true focal length f.
3. Compute the N-vectors of points $P_1, \cdots, P_9$ with respect to the temporal focal length $\hat{f}$ by Eq. (12).
4. Compute the N-vector $\boldsymbol{m}$ of the vanishing point of lines $P_1P_2P_3$, $P_8P_9P_4$, and $P_7P_6P_5$, and the N-vector $\boldsymbol{m}'$ of the vanishing point of lines $P_7P_8P_1$, $P_6P_9P_2$, and $P_5P_4P_3$ with respect to the temporal focal length $\hat{f}$ by the method described in Sect. 5.
5. Determine the true focal length f by Eq. (17).
6. Repeat the above procedure N times, each time changing the position of the camera relative to the calibration board, and take the optimal weighted average $\bar{f}$ as given in Proposition 4.
7. Compute the variance $V[\bar{f}]$ of the optimal estimate $\bar{f}$ by Proposition 4. The true focal length f is estimated to be in the range of $\bar{f} \pm 1.96\sqrt{V[\bar{f}]}$ with 95% confidence. □

The theoretically optimal weights are given in Eq. (19), and the variance $V[f_\alpha]$ can be computed by Eq. (18). However, this computation requires a priori information about the accuracy of pixel data (see Appendix A). This prior knowledge can be dispensed with by an "a posteriori approximation" given in Appendix B. If this approximation is adopted, the variance $V[\bar{f}]$ of Eq. (20) can be approximated by

$$V[\bar{f}] \approx \sum_{\alpha=1}^{N} \frac{(f_\alpha - \bar{f})^2}{V[f_\alpha]^2} \Big/ \left(\sum_{\alpha=1}^{N} \frac{1}{V[f_\alpha]}\right)^2. \quad (21)$$

8. Procedure of Pose Parameter Calibration

Again, we use the same calibration board on which the square grid pattern of Fig. 8 is drawn (Fig. 9). We define the $\bar{X}\bar{Y}\bar{Z}$ scene coordinate system by regarding point P_9 as the origin $\bar{O}$, and define the $\bar{X}$- and $\bar{Y}$-axes by $\boldsymbol{e}_1 = \overrightarrow{P_9P_4}$ and $\boldsymbol{e}_2 = \overrightarrow{P_9P_2}$. The $\bar{Z}$-axis is defined to be perpendicular to both the $\bar{X}$- and Y-axes: $\boldsymbol{e}_3 = \boldsymbol{e}_1 \times \boldsymbol{e}_2$. We use the scale such that the sides of the four squares are all of unit length.

[Calibration of the pose parameters]

1. Detect line segments in the calibration pattern image.
2. Fit lines to the detected line segments, and determine the N-vectors of the fitted lines with respect to the true focal length f.
3. Compute the N-vectors of points $P_1, \cdots, P_9$ by Eq. (12).
4. Compute the N-vector $\boldsymbol{m}$ of the vanishing point of lines $P_1P_2P_3$, $P_8P_9P_4$, and $P_7P_6P_5$, and the N-vector $\boldsymbol{m}'$ of the vanishing point of lines $P_7P_8P_1$, $P_6P_9P_2$, and $P_5P_4P_3$ by the method described in Sect. 5.
5. Adjust the signs of $\boldsymbol{m}$ and $\boldsymbol{m}'$ so that $(\boldsymbol{m}, \boldsymbol{m}_4 - \boldsymbol{m}_8) > 0$ and $(\boldsymbol{m}', \boldsymbol{m}_2 - \boldsymbol{m}_6) > 0$, where $\boldsymbol{m}_2$, $\boldsymbol{m}_4$, $\boldsymbol{m}_6$, and $\boldsymbol{m}_8$ are, respectively, the N-vectors of points P_2, P_4, P_6, and P_8.
6. Compute an orthonormal right-hand system $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$ such that $\boldsymbol{e}_1 \approx \boldsymbol{m}$ and $\boldsymbol{e}_2 \approx \boldsymbol{m}'$ in an optimal manner (Appendix C).
7. Adjust the locations of points $P_0, \cdots, P_9$ so that the projective geometric constraints (see Appendix D) are all satisfied. Let $\bar{P}_0, \cdots, \bar{P}_9$ be the corrected positions.
8. Let $\boldsymbol{m}_o$ be the N-vector of point $\bar{P}_9$, and compute the distance r_o by Eq. (9), in which we take the reference point Q to be any of the eight points $\bar{P}_1, \cdots, \bar{P}_8$ with the knowledge that in the scene $|\bar{P}_9\bar{P}_1|$, $|\bar{P}_9\bar{P}_3|$, $|\bar{P}_9\bar{P}_7|$ are all $\sqrt{2}$, while $|\bar{P}_9\bar{P}_2|$, $|\bar{P}_9\bar{P}_4|$, $|\bar{P}_9\bar{P}_6|$, and $|\bar{P}_9\bar{P}_8|$ are all 1.
9. Compute the pose parameters $\{\boldsymbol{R}, \boldsymbol{h}\}$ by Eqs. (4) and (5). □

Step 1-4 are the same as for the calibration of the focal length f except that the already determined true focal length f is used instead of a tentative value $\hat{f}$. In Step 5, the signs of $\boldsymbol{m}$ and $\boldsymbol{m}'$ are chosen so that they

are respectively oriented along $P_8\vec{P}_4$ and $P_6\vec{P}_2$.

Vectors $\boldsymbol{m}$ and $\boldsymbol{m}'$ computed in Step 4 should the oretically be orthogonal to each other, but they may not be exactly orthogonal in the presence of noise. In Step 6, they are forced to be orthogonal. This is necessary because otherwise the rotation matrix $\boldsymbol{R}$ computed in Step 9 may not be an orthogonal matrix.

In Step 7, the pattern is adjusted so that it can be a perspective projection of an orthogonal square-grid pattern (cf. Appendix D). The choice of the reference point in Step 9 is arbitrary; any choice will yield an identical result due to the consistency enforced in Step 7.

9. Concluding Remarks

We have presented a new theoretical framework for camera calibration using images by representing all points and lines by unit vectors, which we called "N-vectors"[7]. The calibration process is decomposed into atomic modules: The focal length f and the pose parameters $\{\boldsymbol{R}, \boldsymbol{h}\}$ are computed by detecting, on the image plane, the "vanishing points" of two sets of lines that are mutually orthogonal in the scene; the absolute distance of the scene coordinate origin is determined by locating a point whose scene coordinates are known. This decomposition has the following advantages:

- The mathematics is extremely refined in terms of N-vectors in each module.
- Computation steps are given as simple algebraic expressions without iterations.
- Statistical analysis of error behaviors becomes easy, and as a result noise robustness can be maximized.
- Not only statistically optimal estimates but also their "variances" are obtained in analytical forms, from which we can deduce the "confidence level" quasntitatively.

These advantages would be impossible if an overall parametric fitting approach were adopted.

In this paper, image distortions due to lens abberation and raster scanning (including the "aspect ratio") are not considered, because image distortions can be corrected as a separate process beforehand, while the focal length and the pose parameters (and the motion parameters) are of ten changed, necessitating quick recalibrations frequently.

We also assumed that the center of the image plane was known, say at the center of the frame. The exact location of the center of the image plane is not necessary in some applications. It is also reported that a small distortion of its location affects 3-D interpretation of the scene almost negligibly[5]. The exact center of the image is difficult to locate by the very reason that it affects 3-D interpretation very little. This fact makes the overall parametric fitting approach all the more dangerous, because the center of the image is easily distorted by other parameters in the course of optimization.

If we do want to locate the image origin, the most effective way may be the use of a "mechanical" method, since image analysis is very insensitive to the location of the image origin. For example, identifying the faces of the camera body that are parallel to the optical axis of the lens, we move the camera along the assumed optical axis and detect the "focus of expansion" on the image plane[7].

Acknowledgement

The authors thank Michael Brady, Andrew Blake, Roberto Cipolla, David Forsyth, and Andrew Zisserman of the University of Oxford, U. K., for detailed comments and discussions on this work.

References

(1) Caprile B. and Torre V.: "Using vanishing points for camera calibration", Int. J. Comput. Vision, **4**, pp. 127–140 (1990).

(2) Echigo T.: "A camera calibration technique using sets of parallel lines", Machine Vision Appl., **3**, pp. 159–167 (1990).

(3) Faugeras O. D. and Toscani G.: "Camera calibration for 3D computer vision", Proc. Int. Workshop Industrial Appl. Machine Vision Machine Intell., Tokyo, Japan, pp. 240–247 (Feb. 1987).

(4) Grosky W. I and Tamburino L. A.: "A unified approach to the linear camera calibration problem", IEEE Trans. Pattern Anal. Machine Intell., **12**, pp. 663–671 (1990).

(5) Hung Y. -P. and Shieh S. -W.: "When should we consider lens distortion in camera calibration", Proc. IAPR Workshop on Machine Vision Applications, Tokyo, Japan, pp. 367–370 (Nov. 1990).

(6) Kanatani K.: "Group-Theoretical Methods in Image Understanding", Springer, Berlin (1990).

(7) Kanatani K.: "Computational projective geometry", CVGIP: Image Understanding, **54** (1991) (to appear).

(8) Kanatani K.: "Hypothesizing and testing geometric properties of image data", CVGIP: Image Understanding, **54** (1991) (to appear).

(9) Kanatani K. and Onodera Y.: "Camera calibration by computational projective geometry", Proc. IAPR Workshop on Machine Vision Applications, Tokyo, Japan, pp. 363–366 (Nov. 1990).

(10) Kumar R. and Hanson A. R.: "Sensitivity of the pose refinement problem to accurate estimation of camera parameters", Proc, Int. Conf. Comput. Vision, Osaka, Japan, pp. 365–369 (Dec. 1990).

(11) Lentz R. K. and Tsai R. Y.: "Techniques for calibration of the scale factor and image center for high-accuracy 3-D machine vision metrology", IEEE Trans. Pattern Anal. Machine Intell., **10**, pp. 713–720 (1988).

(12) Lentz R. K. and Tsai R. Y.: "Calibrating a Cartesian robot with eye-on-hand configuration independent of eye-to-hand relationship", IEEE Trans. Pattern Anal. Machine Intell., **11**, pp. 916–928 (1989).

(13) Tsai R. Y.: "An efficient and accurate camera calibration technique for 3D machine vision", Proc. IEEE Conf. Comput. Vision Pattern Recog., Miami Beach, FL, U. S.

A, pp. 364-376 (June 1986).

Appendix A

The "covariance matrix" of vector quantity $\boldsymbol{a}$ is defined by

$$V[\boldsymbol{a}]= E\,[\Delta\boldsymbol{a}\,\Delta\boldsymbol{a}^T]. \qquad (A\cdot1)$$

Here, quantity $\boldsymbol{a}$ is assumed to be disturbed into $\boldsymbol{a}+\Delta\boldsymbol{a}$ by random disturbance $\Delta\boldsymbol{a}$ of mean 0, and $E[\cdot]$ designates the expectation.

As shown in the text, the accuracy of the estimation of vanishing points, thereby the focal length and the pose parameters, depends on the accuracy of line fitting, and the accuracy (or rather "inaccuracy") of line fitting is quantitatively described by the covariance matrix $V[\boldsymbol{n}]$ of the N-vector $\boldsymbol{n}$ of the fitted line. It can be theoretically estimated as shown below, but since the proof requires many mathematical preliminaries and long derivations, we only list the finial results.

If a line is fitted to an edge segment, the covariance matrix $V[\boldsymbol{n}]$ of the N-vector $\boldsymbol{n}$ of the fitted line is given by

$$V[\boldsymbol{n}]\approx\frac{\varepsilon^2}{2\gamma w}\left(\frac{12}{w^2}\boldsymbol{u}\boldsymbol{u}^T+\frac{1}{f^2}\boldsymbol{m}_G\boldsymbol{m}_G^T\right), \qquad (A\cdot2)$$

where ε (measured in pixels) is the average displacement of each edge point (assuming that statistical behaviors are the same for all edge points and independent of each other), γ(pixel^{-1}) is the number of edge points per unit length, w (measured in pixels) is the length of the edge segment, $\boldsymbol{u}$ is the unit vector indicating the orientation of the fitted line, $\boldsymbol{m}_G$ is the N-vector of the center of the edge segment[8], and f (measured in pixels) is the value of the focal length used in the computation (it need not be the true focal length).

Appendix B

Consider a projection image of a square grid pattern of Fig. 8. Let $\boldsymbol{m}_1, \cdots, \boldsymbol{m}_9$ be the N-vectors of points $P_1, \cdots, P_9$, respectively. Let $l_1, \cdots, l_6$ be the lines passing through image points $\{P_1, P_2, P_3\}$, $\{P_8, P_9, P_4\}$, $\{P_7, P_6, P_5\}$, $\{P_1, P_8, P_7\}$, $\{P_2, P_9, P_6\}$, $\{P_3, P_4, P_5\}$, respectively. Let $\boldsymbol{n}_1, \cdots, \boldsymbol{n}_6$ be their respective N-vectors. Then, the variance $V[f]$ of Eq. (18) is approximated as follows (we omit the details):

$$V[f]\approx\frac{\text{const.}}{(m_3m_3')^2}\left(\frac{|\boldsymbol{m}\boldsymbol{m}'\boldsymbol{n}_2|^2}{\lambda}+\frac{|\boldsymbol{m}\boldsymbol{m}'\boldsymbol{n}_5|^2}{\lambda'}\right), \qquad (A\cdot3)$$

where

$$\lambda=|P_1P_3|^3\frac{|\boldsymbol{m}\boldsymbol{n}_1\boldsymbol{n}_2|^2}{|\boldsymbol{m}\boldsymbol{n}_1\boldsymbol{m}_2|^2}+|P_7P_5|^3\frac{|\boldsymbol{m}\boldsymbol{n}_2\boldsymbol{n}_3|^2}{|\boldsymbol{m}\boldsymbol{n}_3\boldsymbol{m}_6|^2}, \qquad (A\cdot4)$$

$$\lambda'=|P_1P_7|^3\frac{|\boldsymbol{m}'\boldsymbol{n}_4\boldsymbol{n}_5|^2}{|\boldsymbol{m}'\boldsymbol{n}_4\boldsymbol{m}_8|^2}+|P_3P_5|^3\frac{|\boldsymbol{m}'\boldsymbol{n}_5\boldsymbol{n}_6|^2}{|\boldsymbol{m}'\boldsymbol{n}_6\boldsymbol{m}_4|^2}. \qquad (A\cdot5)$$

Here, $|P_1P_3|$, etc., denote the lengths of the corresponding line segments in the image (measured in pixels), and $|\boldsymbol{abc}|(=(\boldsymbol{a},\boldsymbol{b}\times\boldsymbol{c})=(\boldsymbol{b},\boldsymbol{c}\times\boldsymbol{a})=(\boldsymbol{c},\boldsymbol{a}\times\boldsymbol{b}))$ denotes the scalar triple product of vectors $\boldsymbol{a}$, $\boldsymbol{b}$, and $\boldsymbol{c}$. The constant in Eq.(A·3) can be chosen arbitrarily; it does not affect Eqs. (19) and (21).

Appendix C

Let $\boldsymbol{m}_1$ and $\boldsymbol{m}_2$ be unit vectors with covariance matrices $V[\boldsymbol{m}_1]$ and $V[\boldsymbol{m}_2]$, respectively. The orthonormal system $\{\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3\}$ such that $\boldsymbol{e}_1\approx\boldsymbol{m}_1$ and $\boldsymbol{e}_2\approx\boldsymbol{m}_2$ is optimally determined as follows (we omit the proof):

1. Let $\boldsymbol{M}$ be the matrix consisting of tr$V[\boldsymbol{m}_1]\,\boldsymbol{m}_1$, tr$V[\boldsymbol{m}_2]\,\boldsymbol{m}_2$, and $\boldsymbol{0}$ as its three columns in this order:

$$\boldsymbol{M}=\left[\frac{\boldsymbol{m}_1}{trV[\boldsymbol{m}_1]},\ \frac{\boldsymbol{m}_2}{trV[\boldsymbol{m}_2]},\ \boldsymbol{0}\right] \qquad (A\cdot6)$$

where tr$V[\boldsymbol{m}_1]$ and tr$V[\boldsymbol{m}_2]$ are the traces of covariance matrices $V[\boldsymbol{m}_1]$ and $V[\boldsymbol{m}_2]$, respectively.

2. Let

$$\boldsymbol{M}=\boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{U}^T \qquad (A\cdot7)$$

be the "singular value decomposition" of $\boldsymbol{M}$, where $\boldsymbol{V}$ and $\boldsymbol{U}$ are orthogonal matrices, and $\boldsymbol{\Lambda}$ a diagonal matrix with nonnegative diagonal elements.

3. Vectors $\boldsymbol{e}_1$, $\boldsymbol{e}_2$, and $\boldsymbol{e}_3$ are given as the first, the second, the third columns of matrix $\boldsymbol{V}\boldsymbol{U}^T$, respectively:

$$(\boldsymbol{e}_1,\ \boldsymbol{e}_2,\ \boldsymbol{e}_3)=\boldsymbol{V}\boldsymbol{U}^T. \qquad (A\cdot8)$$

If $\boldsymbol{m}_1$ and $\boldsymbol{m}_2$ are equally reliable and their distributions are approximately isotropic ($V[\boldsymbol{m}_1]\approx V[\boldsymbol{m}_2]\approx$ const. $\times\boldsymbol{I}$), the following computation gives a simple approximation (Fig. A·1):

$$\boldsymbol{e}_1=\frac{1}{\sqrt{2}}(N[\boldsymbol{m}+\boldsymbol{m}']+N[\boldsymbol{m}-\boldsymbol{m}']),$$

$$\boldsymbol{e}_2=\frac{1}{\sqrt{2}}(N[\boldsymbol{m}+\boldsymbol{m}']-N[\boldsymbol{m}-\boldsymbol{m}']),$$

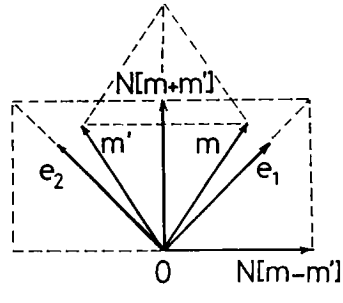

Fig. A·1 Enforcement of orthogonality of two unit vectors.

$$\boldsymbol{e}_3 = \boldsymbol{e}_1 \times \boldsymbol{e}_2. \tag{A·9}$$

Appendix D

Let Fig. A·2 be a perspective projection of the square grid pattern of Fig. 8. Let $\boldsymbol{m}_1, \cdots, \boldsymbol{m}_9$ be the N-vectors of points $P_1, \cdots, P_9$, respectively. Let lines $l_1, \cdots, l_6$ be the lines passing through $\{P_1, P_2, P_3\}$, $\{P_8, P_9, P_4\}$, $\{P_7, P_6, P_5\}$, $\{P_1, P_8, P_7\}$, $\{P_2, P_9, P_6\}$, $\{P_3, P_4, P_5\}$, respectively. Let $\boldsymbol{n}_1, \cdots, \boldsymbol{n}_6$ be their respective N-vectors. Let P and Q be the vanishing points of lines $\{l_1, l_2, l_3\}$ and $\{l_4, l_5, l_6\}$, resectively. Let $\boldsymbol{m}_P$ and $\boldsymbol{m}_Q$ be their respective N-vectors. Define the vanishing line l_∞ as the line passing through the vanishing points P and Q. Let $\boldsymbol{n}_\infty$ be its N-vector.

Define the diagonal lines l_7 and l_8 as the lines passing through points $\{P_1, P_9, P_5\}$ and $\{P_3, P_9, P_7\}$, respectively. Let $\boldsymbol{n}_7$ and $\boldsymbol{n}_8$ be their respective N-vectors. Let R and S be the intersections of line l_∞ with lines l_7 and l_8, respectively. They are, respectively, the vanishing points of lines l_7 and l_8. Let $\boldsymbol{m}_R$ and $\boldsymbol{m}_S$ be their respective N-vectors.

By the above definition of points $P_1, \cdots, P_9$ and lines $l_1, \cdots, l_8$, the following constraints must be satisfied.

1. Points P_1, P_2, and P_3 must be on line l_1:

$$(\boldsymbol{n}_1, \boldsymbol{m}_1) = 0, \quad (\boldsymbol{n}_1, \boldsymbol{m}_2) = 0, \quad (\boldsymbol{n}_1, \boldsymbol{m}_3) = 0. \tag{A·10}$$

2. Points P_8, P_9, and P_4 must be on line l_2:

$$(\boldsymbol{n}_2, \boldsymbol{m}_8) = 0, \quad (\boldsymbol{n}_2, \boldsymbol{m}_9) = 0, \quad (\boldsymbol{n}_2, \boldsymbol{m}_4) = 0. \tag{A·11}$$

3. Points P_7, P_6, and P_5 must be on line l_3:

$$(\boldsymbol{n}_3, \boldsymbol{m}_7) = 0, \quad (\boldsymbol{n}_3, \boldsymbol{m}_6) = 0, \quad (\boldsymbol{n}_3, \boldsymbol{m}_5) = 0. \tag{A·12}$$

4. Points P_1, P_8, and P_7 must be on line l_4:

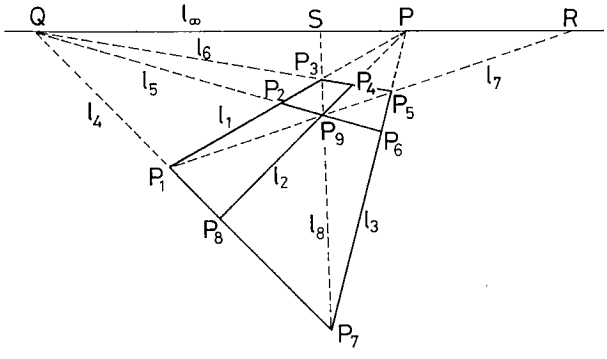

Fig. A·2 A perspective projection image of a square grid pattern.

$$(\boldsymbol{n}_4, \boldsymbol{m}_1) = 0, \quad (\boldsymbol{n}_4, \boldsymbol{m}_8) = 0, \quad (\boldsymbol{n}_4, \boldsymbol{m}_7) = 0. \tag{A·13}$$

5. Points P_2, P_9, and P_6 must be on line l_5:

$$(\boldsymbol{n}_5, \boldsymbol{m}_2) = 0, \quad (\boldsymbol{n}_5, \boldsymbol{m}_9) = 0, \quad (\boldsymbol{n}_5, \boldsymbol{m}_6) = 0. \tag{A·14}$$

6. Points P_3, P_4, and P_5 must be on line l_6:

$$(\boldsymbol{n}_6, \boldsymbol{m}_3) = 0, \quad (\boldsymbol{n}_6, \boldsymbol{m}_4) = 0, \quad (\boldsymbol{n}_6, \boldsymbol{m}_5) = 0. \tag{A·15}$$

7. Points P_1, P_9, and P_5 must be on line l_7:

$$(\boldsymbol{n}_7, \boldsymbol{m}_1) = 0, \quad (\boldsymbol{n}_7, \boldsymbol{m}_9) = 0, \quad (\boldsymbol{n}_7, \boldsymbol{m}_5) = 0. \tag{A·16}$$

8. Points P_3, P_9, and P_7 must be on line l_8:

$$(\boldsymbol{n}_8, \boldsymbol{m}_3) = 0, \quad (\boldsymbol{n}_8, \boldsymbol{m}_9) = 0, \quad (\boldsymbol{n}_8, \boldsymbol{m}_7) = 0. \tag{A·17}$$

From the definition of the vanishing points and the vanishing line, the following constraints must be satisfied.

9. Lines l_1, l_2 and l_3 must meet at point P:

$$(\boldsymbol{m}_P, \boldsymbol{n}_1) = 0, \quad (\boldsymbol{m}_P, \boldsymbol{n}_2) = 0, \quad (\boldsymbol{m}_P, \boldsymbol{n}_3) = 0. \tag{A·18}$$

10. Lines l_4, l_5 and l_6 must meet at point Q:

$$(\boldsymbol{m}_Q, \boldsymbol{n}_4) = 0, \quad (\boldsymbol{m}_Q, \boldsymbol{n}_5) = 0, \quad (\boldsymbol{m}_Q, \boldsymbol{n}_6) = 0. \tag{A·19}$$

11. Points P, Q, R, and S must be on line l_∞:

$$(\boldsymbol{n}_\infty, \boldsymbol{m}_P) = 0, \quad (\boldsymbol{n}_\infty, \boldsymbol{m}_Q) = 0,$$
$$(\boldsymbol{n}_\infty, \boldsymbol{m}_R) = 0, \quad (\boldsymbol{n}_\infty, \boldsymbol{m}_S) = 0. \tag{A·20}$$

From the orthogonality of the lines in the scene, the following constraints must also be satisfied.

12. The vanishing points P and Q must indicate mutually orthogonal 3-D orientations, and the vanishing points R and S must indicate mutually orthogonal 3-D orientations, too:

$$(\boldsymbol{m}_P, \boldsymbol{m}_Q) = 0, \quad (\boldsymbol{m}_R, \boldsymbol{m}_S) = 0. \tag{A·21}$$

Unless all these conditions are satisfied, the pattern cannot be regarded as a perspective projection of a square grid pattern. If the pattern is detected by image processing of a real image, it does not necessarily satisfy these conditions (in particular, Eqs. (A·16) -(A·19) and (A·21)) due to image noise and image distortions. Hence, the computed N-vectors of points and lines must be adjusted so that Eqs. (A·10)-(A·21) are all satisfied. Since each N-vector is coupled with multiple N-vectors, we must resort to iterations. The details of this procedure are omitted.

Kenichi Kanatani was born in Okayama, Japan in 1947, and received his B. E., M. E. and Ph. D. degrees in applied mathematics from the University of Tokyo, Japan in 1972, 1974 and 1979, respectively. In 1979, he joined Gunma University, Japan, where he is now Professor of Computer Science. He studied physics at Case Western Reserve University, USA from 1969 to 1970. He was Visiting Researcher at the University of Maryland, USA. from 1985 to 1986, at the University of Kopenhagen, Denmark in 1988, and at the University of Oxford, U. K. in 1991. He is the author of "Group-Theoretical Methods in Image Understanding" (Springer 1990).

Yasuhiro Onodera was born in Sano, Tochigi, Japan in 1966, and received his B. E and M. E. degrees in computer science from Gunma University, Japan in 1989 and 1991, respectively. In 1991 he joined Atsugi Technology Center, Sony Corporation, Japan. He is currently engaged in broadcast systems design development.