

## Detecting the Motion of a Planar Surface by Line and Surface Integrals

KEN-ICHI KANATANI

*Department of Computer Science, Gunma University, Kiryu, Gunma 376, Japan*

Received December 7, 1983; revised April 30, 1984

A mathematical formulation is presented for detecting the 3D motion of a planar surface from the motion of its perspective image without knowing correspondence of points. The motion is determined explicitly by numerical computation of certain line or surface integrals on the image. The same principle is also used to know the position and orientation of a planar surface fixed in the space by moving the camera or using several appropriately positioned cameras, and no correspondence of points is involved. Some numerical examples are also given.

© 1985 Academic Press, Inc.

### 1. INTRODUCTION

Detecting the 3-dimensional motion of a surface from the motion of its projected image is one of the most important tasks of computer vision and image processing. Kanatani [1, 2] gave a mathematical formulation in the case of a moving planar surface under orthographic projection. In [1], he applied a mathematical principle called "stereology" or "integral geometry" to the texture when the surface is textured, and in [2], he modified the theory so that the contour image alone is used. In both cases, there is no need to know the "correspondence," i.e., the knowledge of "which point moves to which one," and the motion is explicitly given in terms of a small number of data measured on the plane of vision. Related topics and their background are also reviewed there. However, these theories are approximate in the sense that some regular conditions are assumed with respect to the texture or the contour shape.

As is discussed in [2], the correspondence of points is a local characteristic and hence is sensitive to local errors, so that a large number of correspondence pairs are necessary and averaging processes are required to cancel out the local errors. Moreover, the detection of correspondence is usually a time consuming process and is sometimes impossible, especially when the figure is a curve with no edges or corners. Also discussed in [2] is the importance of computation in terms of explicit analytical expressions without resorting to optimization searches by iterative processes (e.g., searching for the best matching of two images, etc.) in view of computational time and convergence problems.

This paper extends the method in [2] to the case of perspective projection and gives a rigorous mathematical formulation which is theoretically exact. In other words, unlike the previous studies, no special assumptions are made on the shape of the image. No correspondence of points is involved, either. The motion is explicitly given by a set of "linear" equations without requiring any iterative processes or matching processes. In the previous two studies [1, 2], we singled out surface rotations, but here translations are also incorporated. The entire formulation turns out to be a special case of Amari's theory of the invariant feature space [3, 4].

We assume, as in [2], that the surface is planar having a closed contour, or equivalently it can be an infinite plane on which a closed curve is drawn. Since we are viewing the image of a closed curve alone, all relevant information must be extracted from the curve. Here, we calculate line or surface integrals of several functions, where integrations are executed numerically. Since they are obtained as sums or averages of a large number of observed data, they are insensitive to local errors in general. Then, we obtain a set of differential equations linear in motion parameters describing the time change of these integrals. The 3-dimensional motion is recovered by solving them. Hence, the correspondence of points or the "optical flow" [5-7] is obtained as a result of this computation. The same principle is also applied to determine the position and orientation of a planar surface fixed in the space by moving the camera or equivalently using several appropriately positioned cameras. The correspondence of points is not required, either. Some numerical examples are also given.

## 2. OPTICAL FLOW INDUCED BY THREE DIMENSIONAL MOTION

Consider an  $xyz$  coordinate system in the space, and let the  $xy$  plane be the plane of vision. Suppose the center of projection is located at  $(0, 0, -l)$ , i.e., at distance  $l$  from the plane on the negative side of the  $z$  axis (Fig. 1). The perspective projection projects a point  $(X, Y, Z)$  in the space to  $(x, y, 0)$  on the plane, where

$$x = lX/(l + Z), \quad y = lY/(l + Z). \quad (2.1)$$

The orthographic projection is attained by taking the limit of  $l \rightarrow \infty$ . Suppose the point  $(X, Y, Z)$  is on a plane whose equation is  $Z = pX + qY + r$ . Parameters  $p$  and  $q$  are known as the "gradients" of the plane, since  $p = \partial Z/\partial X$  and  $q = \partial Z/\partial Y$ . Let us call  $p, q$ , and  $r$  the "surface parameters" of the plane. Suppose the surface is at a given instant translating with instantaneous velocity  $(a, b, c)$  and rotating with instantaneous angular velocity  $(\omega_1, \omega_2, \omega_3)$  about  $(0, 0, r)$ , the intersection between the surface and the  $z$  axis. (Let us call  $a, b, c, \omega_1, \omega_2$ , and  $\omega_3$  the "motion parameters" of the plane.) The instantaneous velocity of point  $(X, Y, Z)$  becomes

$$dX/dt = a + \omega_2(Z - r) - \omega_3Y, \quad (2.2)$$

$$dY/dt = b + \omega_3X - \omega_1(Z - r), \quad (2.3)$$

$$dZ/dt = c + \omega_1Y - \omega_2X. \quad (2.4)$$

In view of Eq. (2.1), this 3-dimensional motion induces a velocity field (or "optical

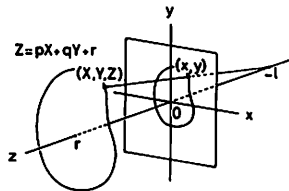


FIG. 1. Relationship between the spatial coordinates  $(X, Y, Z)$  of a point and the plane coordinates  $(x, y)$  of its perspective projection.

flow")  $u = dx/dt$  and  $v = dy/dt$  on the plane of vision in the form

$$u = \frac{l - px - qy}{l + r} \left( a - \frac{x}{l}c \right) - \frac{xy}{l} \omega_1 + \left( px + qy + \frac{x^2}{l} \right) \omega_2 - y \omega_3, \quad (2.5)$$

$$v = \frac{l - px - qy}{l + r} \left( b - \frac{y}{l}c \right) - \left( px + qy + \frac{y^2}{l} \right) \omega_1 + \frac{xy}{l} \omega_2 + x \omega_3. \quad (2.6)$$

If the points on the plane  $Z = pX + qY + r$  move according to Eqs. (2.2)–(2.4), the surface parameters change according to

$$dp/dt = pq\omega_1 - (p^2 + 1)\omega_2 - q\omega_3, \quad (2.7)$$

$$dq/dt = (q^2 + 1)\omega_1 - pq\omega_2 + p\omega_3, \quad (2.8)$$

$$dr/dt = c - pa - qb. \quad (2.9)$$

### 3. INTEGRAL FEATURES AND MOTION DETECTION

Suppose we have a closed curve  $C$  on the plane of vision. Consider as its "features" various line integrals along  $C$  of the form

$$I = \int_C F(x, y) ds, \quad (3.1)$$

where  $ds = \sqrt{dx^2 + dy^2}$ . We must resort to numerical integration for evaluation, and one of the simplest schemes is

$$I \approx \sum_{i=0}^{N-1} F(\bar{x}_i, \bar{y}_i) \Delta s_i, \quad (3.2)$$

$$\Delta x_i = x_{i+1} - x_i, \quad \Delta y_i = y_{i+1} - y_i, \quad \Delta s_i = \sqrt{(\Delta x_i)^2 + (\Delta y_i)^2}, \quad (3.3)$$

$$\bar{x}_i = (x_i + x_{i+1})/2, \quad \bar{y}_i = (y_i + y_{i+1})/2, \quad (3.4)$$

where  $(x_i, y_i)$ ,  $i = 0, 1, \dots, N - 1$ , are consecutive  $N$  points distributed along the curve with sufficiently small intervals and  $(x_N, y_N) = (x_0, y_0)$ . Of course, we could use higher order numerical schemes; see, e.g., [8].

Suppose there is a velocity field  $dx/dt = u(x, y)$ ,  $dy/dt = v(x, y)$  on the plane and the curve moves or "flows" according to this flow. Then, as is well known in calculus, the time difference of the integral of Eq. (3.1) due to the flow field becomes

$$\begin{aligned} \frac{dI}{dt} = \int_C \left[ u \frac{\partial F}{\partial x} + v \frac{\partial F}{\partial y} \right. \\ \left. + \left( \frac{\partial u}{\partial x} \left( \frac{dx}{ds} \right)^2 + \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) \frac{dx}{ds} \frac{dy}{ds} + \frac{\partial v}{\partial y} \left( \frac{dy}{ds} \right)^2 \right) F \right] ds. \end{aligned} \quad (3.5)$$

Let us use abbreviations such as  $F_x = \partial F / \partial x$ ,  $F_y = \partial F / \partial y$ ,  $x' = dx/ds$ , and  $y' = dy/ds$ . If we substitute Eqs. (2.5) and (2.6) in Eq. (3.5), we obtain a linear

equation in  $a, b, c, \omega_1, \omega_2,$  and  $\omega_3,$

$$(3.6) \quad dI/dt = C_1a + C_2b + C_3c + c^4\omega_1 + C_5\omega_2 + C_6\omega_3,$$

where

$$C_1 = -\frac{1+t}{1} \left( p \int [x^2F_x + x^2F] ds + q \int [yF_x + x'y'F] ds - l \int F_x ds \right), \quad (3.7)$$

$$C_2 = -\frac{1+t}{1} \left( p \int [x^2F_x + x'y'F] ds + q \int [yF_x + y^2F] ds - l \int F_y ds \right), \quad (3.8)$$

$$C_3 = \frac{1}{1} \left( p \int [x^2F_x + xyF_y + (2xx'^2 + yx'y' + xy'^2)F] ds \right)$$

$$+ q \int [xyF_x + y^2F_y + (yx'^2 + xx'y' + 2yy'^2)F] ds$$

$$- l \int [xF_x + yF_y + F] ds, \quad (3.9)$$

$$C_4 = -p \int [xF_y + x'y'F] ds - q \int [yF_y + y^2F] ds$$

$$- \frac{1}{1} \int [xyF_x + y^2F_y + (yx'^2 + xx'y' + 2yy'^2)F] ds, \quad (3.10)$$

$$C_5 = p \int [xF_x + x^2F] ds + q \int [yF_x + x'y'F] ds$$

$$+ \frac{1}{1} \int [x^2F_x + xyF_y + (2xx'^2 + yx'y' + xy'^2)F] ds, \quad (3.11)$$

$$C_6 = - \int [yF_x - xF_y] ds. \quad (3.12)$$

The surface motion is detected as follows:

*Step 1.* Give the surface parameters  $p, q,$  and  $r,$  and set  $t \rightarrow 0.$

*Step 2.* Choose 6 or more independent functions  $F_1(x, y), F_2(x, y), \dots,$  and compute the corresponding integrals or "features"  $I_1, I_2, \dots,$  at time  $t.$  Compute also  $C_1, \dots, C_6$  of Eqs. (3.7)-(3.12) for each feature. All the line integrals involved can be computed numerically on the plane of vision. In fact,  $F_x, F_y,$  and  $F$  are given functions, and  $x'$  and  $y'$  can be evaluated, say, by computing  $\Delta x_i/\Delta s_i$  and  $\Delta y_i/\Delta s_i$  of Eqs. (3.3) or by some higher order schemes, see, e.g., [8]. For a given camera,  $t$  is a known constant.

*Step 3.* Observe the contour motion for a short time  $\Delta t,$  and compute the time derivatives  $dI/dt, dI_2/dt, \dots,$  by an appropriate numerical scheme. For example, observe  $I'$  at time  $t + \Delta t$  and compute  $[I'(t + \Delta t) - I'(t)]/\Delta t$  or use some higher order difference schemes with three or more consecutive observations, see, e.g., [8, 9]. *Step 4.* Solve the resulting simultaneous equations of the form of Eq. (3.6) for the features  $I_1, I_2, \dots,$  to obtain the motion parameters  $a, b, c, \omega_1, \omega_2,$  and  $\omega_3.$  Since the equations are linear in the motion parameters, they can be solved

immediately. If the number of equations exceeds 6, we can solve them, say, by the least square error method, which reduces to solving a set of linear equations known as the “normal equations.”

*Step 5.* The surface parameters  $p$ ,  $q$ , and  $r$  at time  $t + \Delta t$  are evaluated from Eqs. (2.7)–(2.9), say by

$$p' \approx p + [pq\omega_1 - (p^2 + 1)\omega_2 - q\omega_3] \Delta t, \quad (3.13)$$

$$q' \approx q + [(q^2 + 1)\omega_1 - pq\omega_2 + p\omega_3] \Delta t, \quad (3.14)$$

$$r' \approx r + [c - pa + qb] \Delta t, \quad (3.15)$$

or by some higher order schemes, see, e.g., [8, 9].

*Step 6.*  $p \leftarrow p', q \leftarrow q', r \leftarrow r', t \leftarrow t + \Delta t$ , and go back to Step 2.

Thus, we can trace the position and orientation of a plane surface, provided the position and orientation are known initially, and no correspondence of points is involved. Determination of the initial orientation and position is discussed later.

#### 4. USE OF SURFACE INTEGRALS AND REDUCTION TO ORTHOGRAPHIC PROJECTION

Line integrals are not the only “features” of closed contour  $C$ . Alternatively, we can take surface integrals over the area  $S$  enclosed by the contour curve  $C$ . Consider a surface integral of the form

$$J = \int_S F(x, y) \, dx \, dy. \quad (4.1)$$

This integral can also be evaluated numerically on the plane of vision. The scheme may not be as simple as for line integrals, especially when the contour has an irregular shape, but there is no essential difficulty. Moreover, this can be converted to line integrals. Note that there always exist two functions  $P(x, y)$  and  $Q(x, y)$  such that  $F(x, y)$  is expressed as

$$F(x, y) = \partial Q / \partial x - \partial P / \partial y. \quad (4.2)$$

Then, due to “Green’s theorem” of vector calculus, integral (4.1) is converted to a line integral

$$J = \int_C [P(x, y) \, dx + Q(x, y) \, dy]. \quad (4.3)$$

If there is a velocity field  $dx/dt = u(x, y)$ ,  $dy/dt = v(x, y)$ , the surface integral of Eq. (4.1) is transformed, as is well known in calculus, by

$$\begin{aligned} \frac{dJ}{dt} &= \int_S \left[ u \frac{\partial F}{\partial x} + v \frac{\partial F}{\partial y} + \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) F \right] dx \, dy \\ &= \int_C [uF \, dy + vF \, dx], \end{aligned} \quad (4.4)$$

Note that, due to Green’s theorem, we can express  $dJ/dt$  as either a surface integral

or a line integral. The subsequent procedures go in parallel with the case of line integrals, so we omit the details.

So far, all the formulations have been based on perspective projection. As was mentioned earlier, they are reduced to the case of orthographic projection by taking the limit of  $l \rightarrow \infty$ . The induced 2-dimensional flow becomes

$$u = a + (px + qy)\omega_2 - y\omega_3, \quad (4.5)$$

$$v = b - (px + qy)\omega_1 + x\omega_3. \quad (4.6)$$

If we let  $l \rightarrow \infty$ ,  $C_3$  of Eq. (3.9) vanishes and hence  $c$  cannot be determined, which is obvious for the orthographic projection ( $c$  does not appear in Eqs. (4.5) and (4.6)). However,  $a$ ,  $b$ ,  $\omega_1$ ,  $\omega_2$ , and  $\omega_3$  can be determined by the same procedure, and the orientation of the surface is traced by using Eqs. (2.7) and (2.8). (An alternative method is described in [2].)

##### 5. DETERMINATION OF SURFACE PARAMETERS BY MOVING THE CAMERA

The method discussed so far traces the motion of a planar surfaces by detecting the translational and rotational velocities successively. This type of trace requires known initial conditions. Apparently, it is impossible to know the position and orientation of a surface only from a single image. One way to circumvent this difficulty is application of "stereoscopic" or "binocular" vision. This usually requires the knowledge of point correspondence, but our principle provides us with a scheme without it. Suppose a planar surface is fixed in the space and the camera can move instead. The motion of the camera causes effects equivalent to the motion of the surface in the opposite direction. Hence, we obtain the following procedure.

*Step 1.* Give an appropriate function  $F(x, y)$  and consider the corresponding feature  $I$  of Eq. (3.1)

*Step 2.* Move the camera along the  $x$ ,  $y$ , and  $z$  axes with its orientation fixed and observe  $C_1$ ,  $C_2$ , and  $C_3$  of Eq. (3.6) directly. For example, suppose the camera moves along the  $x$ -axis by  $\Delta a$  and we observe the difference of  $\Delta I$  of feature  $I$  on the plane of vision. Then,  $C_1$  is evaluated by  $-\Delta I/\Delta a$  if  $\Delta a$  is small. (Note that the plane moves by  $-\Delta a$  relative to the camera). Of course, we could also use higher order schemes using several data observed along the camera motion (see, e.g., [8, 9]).

*Step 3.* Compute the surface parameters  $p$ ,  $q$ , and  $r$  by regarding Eqs. (3.7), (3.8), and (3.9) as a set of simultaneous equations with  $p$ ,  $q$ , and  $r$  unknowns. First,  $p$  and  $q$  are given by solving simultaneous equations

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad (5.1)$$

where

$$\begin{aligned} a_{11} = & lC_3 \int_C [xF_x + x'^2F] ds \\ & + C_1 \int_C [x^2F_x + xyF_y + (2xx'^2 + yx'y' + xy'^2)F] ds, \end{aligned} \quad (5.3)$$

$$\begin{aligned}
 a_{12} = & lC_3 \int_C [yF_x + x'y'F] ds \\
 & + C_1 \int_C [xyF_x + y^2F_y + (yx'^2 + xx'y' + 2yy'^2)F] ds, \quad (5.4)
 \end{aligned}$$

$$\begin{aligned}
 a_{21} = & lC_3 \int_C [xF_y + x'y'F] ds \\
 & + C_2 \int_C [x^2F_x + xyF_y + (2xx'^2 + yx'y' + xy'^2)F] ds, \quad (5.5)
 \end{aligned}$$

$$\begin{aligned}
 a_{22} = & lC_3 \int_C [yF_y + y'^2F] ds \\
 & + C_2 \int_C [xyF_x + y^2F_y + (yx'^2 + xx'y' + 2yy'^2)F] ds, \quad (5.6)
 \end{aligned}$$

$$b_1 = l \left( lC_3 \int_C F_x ds + C_1 \int_C [xF_x + yF_y + F] ds \right), \quad (5.7)$$

$$b_2 = l \left( lC_3 \int_C F_y ds + C_2 \int_C [xF_x + yF_y + F] ds \right). \quad (5.8)$$

Then,  $r$  is obtained by substituting  $p$  and  $q$  in one of Eqs. (3.7), (3.8), and (3.9).

The same process is possible if a surface integral is used. In practice, however, a number of different features should be taken to determine the surface parameters  $p$ ,  $q$ , and  $r$ , say, by the least square error method. Here, the camera is moved in three different directions, but it is possible in principle to do the same thing by moving it in only one direction, say along the  $x$  axis, and measuring three or more different features. However, this does not seem suitable in view of error sensitivity and possible degeneracy of the equations. In the above, we used the expression "moving the camera" to emphasize the relationship completely dual to the motion of the plane. Of course, we need not move anything if we have an appropriate number of cameras suitable positioned beforehand. This is a more realistic situation, and data are obtained simultaneously and processed in one stage.

## 7. DISCUSSIONS AND NUMERICAL EXAMPLES

Several problems remain in actual implementation. In order to use line or surface integrals as features, we must choose an appropriate set of 6 or more integrand functions. They should be at least linearly independent, but it is not a sufficient condition. They should be such that the rates of change of the corresponding features are linearly independent with respect to motion parameters. In other words, if we regard the  $C$ s of Eqs. (3.7)–(3.12) as a vector  $(C_1, C_2, C_3, C_4, C_5, C_6)$ , we have as many vectors as the integrands we choose. These vectors must contain at least 6 linearly independent ones, but the necessary and sufficient condition for it has not yet been known. Moreover, the determinant of the matrix composed of those 6 independent vectors must be as large as possible from the viewpoint of computational stability. At present, however, not much is known about the favorable choice of the features.

Figure 2 shows a synthetic image  $C$  of a moving planar surface at time  $t$  and  $C'$  at  $t + \Delta t$  a short time later. The surface parameters are  $p = -0.30$ ,  $q = -1.71$  and  $r/l = 1.00$ . We can determine the motion parameters  $a$ ,  $b$ ,  $c$ ,  $\omega_1$ ,  $\omega_2$ , and  $\omega_3$  by

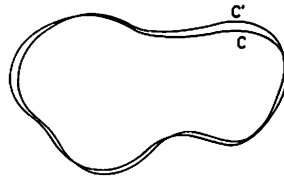


FIG. 2. A perspective image of a planar surface contour at one time  $C$  and its image a short time later  $C'$ .

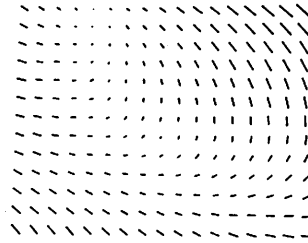


FIG. 3. The optical flow computed from the contour motion of Fig. 2. The correspondence of points is obtained as a result.

adopting, say,  $x$ ,  $y$ ,  $x^2$ ,  $xy$ ,  $y^2$ , and  $x^2y^2$  as the integrands for line integrals, and the orientation and position of  $C'$  is obtained from Eqs. (3.13)–(3.15). Taking 72 points on each curve and executing numerical computation, we obtained  $a \Delta t/l = 0.027$ ,  $b \Delta t/l = 0.027$ ,  $c \Delta t/l = 0.046$ ,  $\omega_1 \Delta t = 0.031$ ,  $\omega_2 \Delta t = 0.031$ , and  $\omega_3 \Delta t = 0.027$ , while the true values are 0.03, 0.03, 0.05, 0.03, 0.03, and 0.03, respectively. The computed orientation and position of  $C'$  become  $p = -0.27$ ,  $q = -1.61$ , and  $r/l = 1.10$ , while the true values are  $p = -0.17$ ,  $q = -1.61$ , and  $r/l = 1.11$ . This is a very good result in view of the rough scheme of computation. Figure 3 shows the computed “optical flow” of the motion in Fig. 2. (The true optical flow is not drawn, since the difference is very small on the figure.) Thus, the correspondence of points is computed as a consequence. Theoretically, the accuracy increases as more points are

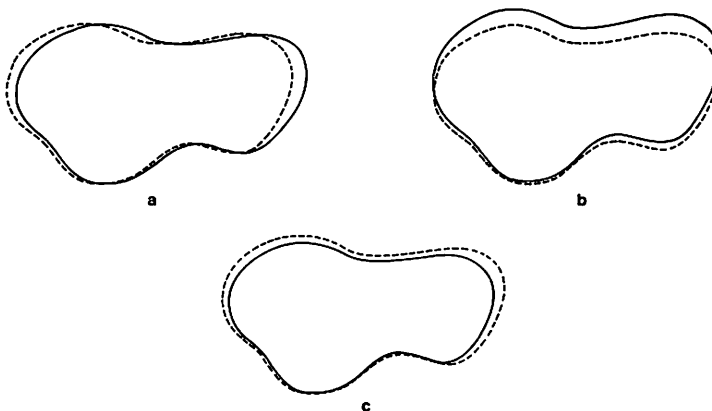


FIG. 4. The image  $C$  of Fig. 2 (dashed) and images of the same surface from different viewpoints (solid). The camera is displaced (a) leftwards, (b) downwards, and (c) away from the surface.



taken on the curves, observations are made at shorter time intervals and higher order schemes are used.

The surface parameters of a fixed plane surface are determined by moving the camera. Figure 4 shows simulated motions of the image when the camera is displaced (a) leftward, (b) downward, and (c) away from the surface. (It is the same surface of Fig. 2, so that  $p = -0.30$ ,  $q = -1.71$ , and  $r/l = 1.00$ .) Taking 72 points on each curve and using the curve length  $\int_C ds$  as the feature, we obtained  $p = -0.36$ ,  $q = -1.93$ , and  $r/l = 1.22$ , a fairly good result. The accuracy increases if we take smaller values of  $\Delta a$ ,  $\Delta b$ , and  $\Delta c$  (as long as the measurement is accurate) or higher order schemes are used. For example, if we make each of  $\Delta a$ ,  $\Delta b$ , and  $\Delta c$  a tenth the above, obtain  $p = -0.30$ ,  $q = -1.73$ , and  $r/l = 1.02$ .

## 8. CONCLUSIONS

The principle of motion detection discussed in this paper has salient characteristics. First, no correspondence of points is required. Second, all we have to do is solve a set of "linear" equations and no iterative search is involved. Third, once we have computed line or surface integrals as the features, we do not have to retain the image in the memory. Only the computed values of the integrals must be stored, because we never compare two images to seek any kind of matching. Fourth, integrals are computed as sums or averages of a large number of observed data, so that the use of integrals as features is insensitive to local errors in general.

It is true that, if correspondence happens to be available easily, we should make use of that information. Indeed, we should combine all available information. For example, since the tracing of motion is performed successively, small errors in each stage may accumulate in the course of time. Hence, it is a wise policy to transform now and then the original image to the one predicted by the computation and check if there is any significant discrepancy between the present image and the computed one, making corrections if necessary. On the other hand, the accuracy of detection increases as the difference of the images become smaller. If the difference is large, one may first predict the motion by our method as a first approximation, transform the first image according to the computed motion and compare it with the second one. The difference should be smaller, so that one can add a more precise correction this time. If it is still unsatisfactory, the procedure is repeated.

In this way, our method shows its merit when combined or used to supplement other measurements. We presented in this paper only synthetic examples, because the performance of our method is greatly affected by many additional techniques, which also include the extraction of surface contours and the method of numerical computation, in particular the use of higher order schemes (see, e.g., [10, 11] for analyses of numerical schemes and errors). Various modifications and variations of our method are possible depending on the apparatus used and the purpose of the analysis.

## ACKNOWLEDGMENTS

Part of this work was done under the support of Saneyoshi Scholarship Foundation.

## REFERENCES

1. K. Kanatani, Detection of surface orientation and motion from texture by a stereological technique, *Artificial Intelligence* **23**, 1984, 213–237.
2. K. Kanatani, Tracing planar surface motion from projection without knowing correspondence, *Comput. Vision Graphics Image Process.* **29**, 1985, 1–12.
3. S. Amari, Invariant structures of signal and feature spaces in pattern recognition problems, *RAAG Memoirs* **4**, 1968, 553–556.
4. S. Amari, Feature spaces which admit and detect invariant signal transformations, in *Proc. 4th Int. Joint Conf. Pattern Recognition*, 1978, pp. 452–456.
5. H. C. Longuet-Higgins and K. Prazdny, The interpretation of a moving retinal image, *Proc. Roy. Soc. Lond. Ser. B* **208**, 1980, 385–397.
6. K. Prazdny, Egomotion and relative depth map from optical flow, *Biol. Cybernetics* **36**, 1980, 87–102.
7. B. K. P. Horn and B. G. Schunk, Determining optical flow, *Artificial Intelligence* **17**, 1981, 185–203.
8. P. Henrici, *Elements of Numerical Analysis*, Wiley, New York, 1964.
9. P. Henrici, *Discrete Variable Methods in Ordinary Differential Equations*, Wiley, New York, 1962.
10. K. Kanatani, The accuracy and preservation property of the discrete mechanics, *Comput. Phys.*, **53**, 1984, 181–187.
11. K. Kanatani, Errors of the incremental method for curves, *Comput. Vision Graphics Image Process.* **26**, 1984, 130–133.