コンピュータビジョンと
イメージメディア 102－4
（1996．11．21）

投票による運動の分類

井宮 淳, イリス フェルミン, 市川 熹

千葉大学　情報工学科
263 千葉稲毛区弥生町 1-33
Tel. 043-290-3257
imiya@ics.tj.chiba-u.ac.jp, firis@icsd4.tj.chiba-u.ac.jp

あらまし　剛体およびその運動が平面的か空間的かを判定するための, ランダム標本化と投票に基づく算法を提案する. また, 既に提案した投票による 3 次元運動解析手法を 2 次元に限った算法を提案する. この算法は平面図形の一部に遮断があっても正しく運動変数を抽出する. また, これらの算法の数値安定性を解析し, 共に誤差に強い算法であることを示す.

キーワード: 運動変数, 確率算法, 運動平面, 平面性, 投票過程

Detection of Planarity and Motion

Atsushi Imiya, Iris Fermin, and Akira Ichikawa

Dept. of Information and Computer Sciences
Chiba University
1-33 Yayoi-cho, Inage-ku, 263 Chiba, Japan
Tel. +81-43-290-3257
imiya@ics.tj.chiba-u.ac.jp, firis@icsd4.tj.chiba-u.ac.jp

Abstract　The recognition of planar shapes is an important problem in computer vision and many methods have been proposed for the analysis of planar motion and spatial motion. Thus, for the analysis of motion we must first determine whether the motion is planar or spatial. Here we propose a randomized algorithm for planarity detection and planar motion estimation.

Key Words: motion parameters, randomized algorithm, planar motion, planarity, voting process

1. Introduction

The recognition of planar polygons in three-dimensional space is an important problem in computer vision. This is partially due to the fact that many three-dimensional objects contain planar polygons [1, 2]. For instance, in object recognition of man-made environments, planar polygons are often present. Similarly, detecting the three-dimensional motion of a surface from the two-dimensional motion of its projected image is a common problem for image analysis. Here the difficulty is how to deal with the point correspondence problem in a sequence of images [3, 4].

In this paper, first we propose a randomized method for the determination of whether an object and its motion are planar. Second we propose a method for planar motion estimation without any assumption about point correspondences. Previously, we proposed a randomized method for three-dimensional motion detection for polyhedral objects [5]. Here the two-dimensional version of that algorithm is presented.

After testing for planarity, we can decide whether the algorithm for the two- or the three-dimensional what is appropriate. Thus, in the first part of this paper we introduce a randomized method to check whether the object is planar. The algorithm is based on the fact that all samples of planar objects moving on a plane lie on the same plane. Planar motion is defined as a planar shape moving on a plane. This means that the shape is rotated about an axis perpendicular to the plane and translated on the same plane. The latter part of the paper concerns a randomized method for planar motion estimation. Henceforth, we assume that the planar object is a polygon and the vertices of the polygon can be determined using an appropriate method. Selecting a triplet of vertices from each image frame, and computing a pair of bases, the motion parameters are estimated by a voting process. In each iteration the rotation matrix is computed and if a match is found, then the score is incremented in a defined voting space. The rotation matrix is determined if a solution with the maximum number of votes is detected in this voting space. After the rotation matrix is estimated the translation vector is determined by randomly selecting points from the two image frames and applying minimization and voting processes. This algorithm is suitable for motion detection for a partially occluded object.

2. Detection of Planarity

In this section we introduce a randomized approach to determine whether an object and its motion are planar.

Given a reference coordinate system x, y and z in three-dimensional Euclidean space $\mathbf{R}^3$ and knowing the coordinate values for three noncollinear points with coordinates $\boldsymbol{x}_i=(x_i, y_i, z_i)^\top$ for i = 1,2,3, if

$$\begin{vmatrix} x & x_1 & x_2 & x_3 \\ y & y_1 & y_2 & y_3 \\ z & z_1 & z_2 & z_3 \\ 1 & 1 & 1 & 1 \end{vmatrix} = 0, \tag{1}$$

then the point $\boldsymbol{x} = (x, y, z)^\top$ lies on the plane defined by the points $\boldsymbol{x}_i$, i= 1,2,3.
Setting

$$\begin{aligned} a &= \begin{vmatrix} y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \\ 1 & 1 & 1 \end{vmatrix} \quad b = -\begin{vmatrix} x_1 & x_2 & x_3 \\ z_1 & z_2 & z_3 \\ 1 & 1 & 1 \end{vmatrix} \\ c &= \begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ 1 & 1 & 1 \end{vmatrix} \quad d = -\begin{vmatrix} x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \\ z_1 & z_2 & z_3 \end{vmatrix}, \end{aligned} \tag{2}$$

eq. (1) becomes

$$ax + by + cz + d = 0. \tag{3}$$

For $\lambda \neq 0$, $(\lambda a, \lambda b, \lambda c, \lambda d)^\top$ determines the same plane. This implies that we can normalize the length of the vector $(a, b, c, d)^\top$ to unity. Therefore, setting

$$\alpha = \frac{a}{\sqrt{a^2+b^2+c^2+d^2}} \quad \beta = \frac{b}{\sqrt{a^2+b^2+c^2+d^2}}$$
$$\gamma = \frac{c}{\sqrt{a^2+b^2+c^2+d^2}} \quad \delta = \frac{d}{\sqrt{a^2+b^2+c^2+d^2}} \tag{4}$$

a vector $(\alpha, \beta, \gamma, \delta)^\top$ determines a point on S^3_+, the positive semi-sphere in four-dimensional Euclidean space $\mathbf{R}^4$. Assuming that we have one image frame and the vertices of the polyhedron have been determined, the following is a summary of the steps for determination of whether an object is planar.

Checking for Planarity of a Shape

1. Randomly select three noncollinear points from the set of image points.
2. Compute the parameters α, β, γ, and δ.
3. Vote for each solution in the voting space.
4. After a sufficient number of iterations, if there is a unique peak then the object is planar.

The above algorithm can be extended for the case when we have two image frames. Thus, the planarity of the object and planar motion are detected simultaneously if a peak is detected for both image frames.

Checking for the Planarity of an Object and Its Motion

1. Randomly select a triplet of points from each image frame.
2. Compute the parameters α, β, γ, and δ for each image frame.
3. Vote for each solution in the voting space.
4. After a sufficient number of iterations, if there is a unique peak then the object and the motion are planar.

3. Planar Motion Estimation

We assume that a planar shape is a polygon and that the vertices of the polygon have been determined in a preprocessing stage. Given a coordinate system x-y. Let $\{\boldsymbol{x}_i\}_{i=1}^n$ be a point set which is moving on a plane, where $\boldsymbol{x}_i = (x_i, y_i)^\top$. For $\boldsymbol{x}_i$, and $\boldsymbol{x}_k$, we set

$$\boldsymbol{x}_{ik} = (x_i - x_k, y_i - y_k)^\top \tag{5}$$

to be invariant under translation. Furthermore,

$$\boldsymbol{x}_{ik}^\perp = (y_i - y_k, -(x_i - x_k))^\top \tag{6}$$

is orthogonal to $\boldsymbol{x}_{ik}$. Thus

$$\boldsymbol{\xi}_{ik} = \frac{\boldsymbol{x}_{ik}}{|\boldsymbol{x}_{ik}|} \quad \boldsymbol{\xi}_{ik}^\perp = \frac{\boldsymbol{x}_{ik}^\perp}{|\boldsymbol{x}_{ik}^\perp|} \tag{7}$$

form an orthogonal basis. Setting

$$\boldsymbol{y}_{\sigma(i)} = \boldsymbol{R}\boldsymbol{x}_i + \boldsymbol{t}, \tag{8}$$

where $\sigma(i)$ is a permutation over $1 \le i \le n$, we define in the same way $\boldsymbol{y}_{ik}$, $\boldsymbol{y}_{ik}^\perp$, $\boldsymbol{\eta}_{ik}$, $\boldsymbol{\eta}_{ik}^\perp$ for the second image frame. Thus, we obtain

$$\begin{aligned} \boldsymbol{x}_{jk} &= \alpha_{ik}^j \boldsymbol{\xi}_{ik} + \alpha_{ik}^{j\perp} \boldsymbol{\xi}_{ik}^\perp \\ \boldsymbol{y}_{jk} &= \beta_{ik}^j \boldsymbol{\eta}_{ik} + \beta_{ik}^{k\perp} \boldsymbol{\eta}_{ik}^\perp, \end{aligned} \tag{9}$$

where

$$\begin{aligned} \alpha_{ik}^j &= \boldsymbol{\xi}_{ik}\boldsymbol{x}_{jk}^\top \quad \alpha_{ik}^{j\perp} = \boldsymbol{\xi}_{ik}^\perp \boldsymbol{x}_{jk}^\top \\ \beta_{ik}^j &= \boldsymbol{\eta}_{ik}\boldsymbol{y}_{jk}{}^\top \quad \beta_{ik}^{j\perp} = \boldsymbol{\eta}_{ik}^\perp \boldsymbol{y}_{jk}{}^\top. \end{aligned} \tag{10}$$

Therefore, if

$$\boldsymbol{y}_\gamma = \boldsymbol{R}\boldsymbol{x}_\gamma + \boldsymbol{t}, \quad \gamma = i, j, k, \tag{11}$$

then

$$\alpha_{ik}^j = \beta_{ik}^j \quad \alpha_{ik}^{j\perp} = \beta_{ik}^{j\perp}. \tag{12}$$

This leads to the three-dimensional motion detection algorithm that we proposed in previous work [5]. Using this algorithm first the rotation matrix is determined and then the translation vector can be determined by applying a minimization process.

1. Rotation Matrix Estimation

1.1 Randomly select three noncollinear points $\{\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k\}$ and $\{\boldsymbol{y}_i, \boldsymbol{y}_j, \boldsymbol{y}_k\}$ from each image frame.

1.2 Compute the vectors $\{\boldsymbol{x}_{ik}, \boldsymbol{x}_{ik}^\perp\}$ and $\{\boldsymbol{y}_{ik}, \boldsymbol{y}_{ik}^\perp\}$ from each image frame.

1.3 Compute the bases $\{\boldsymbol{\xi}_{ik}, \boldsymbol{\xi}_{ij}^\perp\}$, $\{\boldsymbol{\eta}_{ik}, \boldsymbol{\eta}_{ij}^\perp\}$.

1.4 Compute the parameters α_{ik}^j, $\alpha_{ik}^{j\perp}$ and β_{ik}^j, $\beta_{ik}^{j\perp}$.

1.5 If

$$(\alpha_{ik}^j - \beta_{ik}^j)^2 + (\alpha_{ik}^{j\perp} - \beta_{ik}^{j\perp})^2 \le \epsilon^2 \tag{13}$$

then

$$\boldsymbol{R}_E = \left[\boldsymbol{x}_{ik} \;\; \boldsymbol{x}_{ik}^\perp\right] \left[\boldsymbol{y}_{ik} \;\; \boldsymbol{y}_{ik}^\perp\right]^{-1}. \tag{14}$$

1.6 Increase by one the accumulator for the rotation in the voting space.

1.7 After a sufficient number of iterations, if a peak is detected in the voting space for the parameters of the rotation matrix, then the rotation has determined. Otherwise go to step 1.1.

2. Translation Vector Estimation

2.1 Randomly select a triplet of points $\{\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k\}$ and $\{\boldsymbol{y}_i, \boldsymbol{y}_j, \boldsymbol{y}_k\}$

2.2 To guarantee some correspondences we apply the constraint

$$\|\boldsymbol{x}_{\gamma k}| - |\boldsymbol{x}'_{\gamma k}\| \le \epsilon_2 \tag{15}$$

and compute the rotation matrix

$$\boldsymbol{R}_T = [\boldsymbol{x}_{ik} \;\; \boldsymbol{x}_{jk}][\boldsymbol{y}_{ik} \;\; \boldsymbol{y}_{jk}]^{-1}. \tag{16}$$

2.3 If

$$\|\boldsymbol{R}_E - \boldsymbol{R}_T\| \to \min, \tag{17}$$

where $\boldsymbol{R}_E$ is the rotation matrix estimated in the previous procedure, then the translation can be computed using

$$\boldsymbol{t} = \frac{1}{3}\sum_{\gamma=1}^{3}(\boldsymbol{y}_\gamma - \boldsymbol{R}_E\boldsymbol{x}_\gamma). \tag{18}$$

2.4 Increase by one the accumulator for $\boldsymbol{t}$ in the voting space.

2.5 If a threshold in the voting space ($\boldsymbol{t}$) is reached, then stop; otherwise go to step 2.1.

Step 2.2 means that the pair of triplets $\{\boldsymbol{x}_\gamma\}_{\gamma=1}^3$ and $\{\boldsymbol{y}_\gamma\}_{\gamma=1}^3$ should form a congruent triangle within a given predetermined numerical error.

4. Framework for Motion Analysis

Here we introduce a framework for motion analysis based on our randomized approaches for motion estimation in three- and two-dimensional Euclidean spaces without assuming point correspondences. Figure 1 graphically shows the complete methodology for randomized algorithms for motion detection and shape reconstruction.

We can observe in Fig. 1 that assuming we have a set of image views, the three-dimensional object positions can be determined, and then, depending on whether the shape and motion are planar, the algorithm for the estimation of motion is selected. Finally, after motion is estimated the object is reconstructed from motion.

5. Numerical Analysis

In this section we analyze the expected error for both planarity checking and planar motion estimation. These errors are mainly caused by the image digitalization [6, 7].

5.1 Planarity Checking Analysis

Here the errors in the estimation of the plane parameters $(\alpha,\beta,\gamma,\delta)^\top$, which are due to the errors in the estimation of $\boldsymbol{x}_i$ are analyzed. Thus, for

$$\begin{aligned}\boldsymbol{x}_1 &= (x_1,y_1,z_1)^\top \quad \boldsymbol{x}_2=(x_2,y_2,z_2)^\top\\ \boldsymbol{x}_3 &= (x_3,y_3,z_3)^\top,\end{aligned} \tag{19}$$

and

$$\boldsymbol{x}_1' = \boldsymbol{x}_1+\boldsymbol{\epsilon}_1 \quad \boldsymbol{x}_2' = \boldsymbol{x}_2+\boldsymbol{\epsilon}_2 \quad \boldsymbol{x}_3' = \boldsymbol{x}_3+\boldsymbol{\epsilon}_3 \tag{20}$$

such that

$$\frac{|\boldsymbol{\epsilon}_i|}{|\boldsymbol{x}_i|} \ll \omega \quad |\boldsymbol{\epsilon}_i| \le \upsilon, \tag{21}$$

for i= 1,2,3. Setting

$$\begin{aligned} a' &= \begin{vmatrix} y_1' & y_2' & y_3' \\ z_1' & z_2' & z_3' \\ 1 & 1 & 1 \end{vmatrix} \quad b' = -\begin{vmatrix} x_1' & x_2' & x_3' \\ z_1' & z_2' & z_3' \\ 1 & 1 & 1 \end{vmatrix} \\ c' &= \begin{vmatrix} x_1' & x_2' & x_3' \\ y_1' & y_2' & y_3' \\ 1 & 1 & 1 \end{vmatrix} \quad d' = -\begin{vmatrix} x_1' & x_2' & x_3' \\ y_1' & y_2' & y_3' \\ z_1' & z_2' & z_3' \end{vmatrix}, \end{aligned} \tag{22}$$

and

$$\alpha' = \frac{a'}{\sqrt{a'^2+b'^2+c'^2+d'^2}}$$

$$\beta' = \frac{b'}{\sqrt{a'^2+b'^2+c'^2+d'^2}}$$

$$\gamma' = \frac{c'}{\sqrt{a'^2+b'^2+c'^2+d'^2}}$$

$$\delta' = \frac{d'}{\sqrt{a'^2+b'^2+c'^2+d'^2}}, \tag{23}$$

we obtain the following theorem.

Theorem 1

$$(\alpha-\alpha')^2+(\beta-\beta')^2+(\gamma-\gamma')^2+(\delta-\delta')^2 \cong 0. \tag{24}$$

This theorem indicates that the computation of $(\alpha,\beta,\gamma,\delta)^\top$ is stable against numerical and digitalization errors. The proof of this theorem is shown in the appendix.

5.2 Motion Analysis

This section concerns analysis of the expected error in the estimation of the motion parameters $\boldsymbol{R}$ and $\boldsymbol{t}$, which are due to errors in the estimation of $\boldsymbol{x}_\gamma$ and $\boldsymbol{y}_\gamma$. Here, we assume that

$$\boldsymbol{x}_\gamma' = \boldsymbol{x}_\gamma+\boldsymbol{\epsilon}_\gamma \quad \boldsymbol{y}_\gamma' = \boldsymbol{y}_\gamma+\boldsymbol{\sigma}_\gamma \tag{25}$$

are measured, where $\boldsymbol{\epsilon}_\gamma$ and $\boldsymbol{\sigma}_\gamma$ are independent stochastic vectors such that

$$|\boldsymbol{\epsilon}_\gamma| \le \Delta \quad |\boldsymbol{\sigma}_\gamma| \le \Delta \tag{26}$$

for a given small parameter Δ, which depends on the digitalization procedure. Hence, we obtain the following theorems.

Theorem 2 *Let $\boldsymbol{\xi}_{ik}'$ and $\boldsymbol{\eta}_{ik}'$ be a pair of orthogonal bases computed as indicated in the previous section. Then the relations*

$$\begin{aligned}\boldsymbol{\xi}_{ik}' &= \boldsymbol{\xi}_{ik}+a_i\boldsymbol{\delta}_i \qquad \boldsymbol{\eta}_{ik}' = \boldsymbol{\eta}_{ik}+b_i\boldsymbol{\zeta}_i\\ \boldsymbol{\xi}_{ik}'^{\perp} &= \boldsymbol{\xi}_{ik}^{\perp}+a_i^{\perp}\boldsymbol{\delta}_i^{\perp} \qquad \boldsymbol{\eta}_{ik}'^{\perp} = \boldsymbol{\eta}_{ik}^{\perp}+b_i^{\perp}\boldsymbol{\zeta}_i^{\perp}\end{aligned} \tag{27}$$

hold, where $\boldsymbol{\delta}_\gamma$ and $\boldsymbol{\zeta}_\gamma$ are independent random vectors such that

$$|\boldsymbol{\delta}_\gamma| \le 2\Delta \quad |\boldsymbol{\zeta}_\gamma| \le 2\Delta \tag{28}$$

and

$$a_\gamma = \frac{1}{|\boldsymbol{x}_{\gamma k}'|} \quad b_\gamma = \frac{1}{|\boldsymbol{y}_{\gamma k}'|}. \tag{29}$$

Theorem 3 *Let a and b be the maximum values of a_γ and b_γ, respectively. The inequalities*

$$\|\boldsymbol{R}'-\boldsymbol{R}\| \le 2\sqrt{2}(a+b)\Delta \tag{30}$$

$$|r_{ij}'-r_{ij}| \le 2(a+b)\Delta \tag{31}$$

hold, where $\|\boldsymbol{A}\|$ is the matrix norm of a matrix $\boldsymbol{A}$.

Theorem 4 *For a translation vector, the relations*

$$|t' - t| \leq 2\{1 + (a+b)|\bar{x}|\}\Delta \tag{32}$$

and

$$|t'_i - t_i| \leq 2\{1 + (a+b)|\bar{x}^*|\}\Delta \tag{33}$$

hold, where

$$\bar{x} = \frac{1}{n+1}\sum_{i=0}^{n} x_i \tag{34}$$

and x^* is the maximum element of $\bar{x} = (\bar{x_1}, \bar{x_2})$.

Theorem 5 *For a pair of points y_i and y'_i, x_i and x'_i, setting*

$$\alpha^j_{ik} = x^\top_{ik}\xi_{ik} \quad \alpha'^j_{ik} = x'^\top_{ik}\xi'_{ik} \tag{35}$$

and

$$\beta^j_{ik} = y^\top_{ik}\eta_{ik} \quad \beta'^j_{ik} = y'^\top_{ik}\eta'_{ik} \tag{36}$$

the relations

$$\alpha'^j_{ik} = \alpha^j_{ik} + a_i\Delta|x_{ik}| \quad \beta'^j_{ik} = \beta^j_{ik} + b_i\Delta|y_{ik}| \tag{37}$$

and

$$(\alpha'^j_{ik} - \beta'^j_{ik})^2 \cong (\alpha^j_{ik} - \beta^j_{ik})^2 \tag{38}$$

hold.

The Theorems 2, 3, 4, and 5 are the two-dimensional versions of the corresponding versions that we obtained for the three-dimensional case [5]. Equations (30) and (31) give the error for the estimated rotation matrix which depends on the image quantization, and Eqs. (32) and (33) give the error for the estimated translation vector. Theorems 3 and 4 imply that by increasing the resolution which is achieved by decreasing Δ, $\boldsymbol{R}$ and $\boldsymbol{t}$ converge to the true solutions. Furthermore, the relations between r'_{ij} and r_{ij}, and t'_i and t_i show that the resolution of the accumulator space also depends on the resolution of the image. Moreover, the resolution of r'_{ij} depends on the size of the object since constants a and b depend on the size of the object. The resolution of t_i depends on the size of the region of interest. However, in applications the order of a, b and $|x^*|$ is constant. Thus in Theorem 4, we can replace $(a+b)|\bar{x}|$ and $(a+b)|x^*|$ by constants which do not depend on $|\bar{x}|$ and x^*.

Since $|a| \ll 1$ and $|b| \ll 1$, Theorem 3 also implies that the size of the cells of the voting space for the detection of the rotation matrix should be smaller than Δ. However, the size of the cells for the detection of the translation vector depends on the size of the region of interest, because $|x^*|$ depends on the size of the region of interest. Finally, Theorem 5 shows that our criterion for the detection of matching of vectors does not depend on the size of the object and is stable against numerical errors. All the proofs of these theorems are shown in our previous work [5].

5.3 Complexities of Algorithm

For a polygon with n vertices, the number of iterations of the algorithm is $O(n^2)$. In each iteration the algorithm computes a pair coefficients from two images. Thus, the total time complexity is $O((n^2)^2)$. The order of the domain complexity of this algorithm is equivalent to the size of the voting space. Therefore, the domain complexity is $O(n^2)$ because there are $3!(n \times n)$ combinations of points for the two image frames. Thus, the product of the time complexity and the domain complexity of the algorithm is $O(n^6)$.

The probability of choosing any vertex in an image frame is $\frac{1}{n}$, and once a vertex is determined the other three connective vertices are determined. There are 3! combinations for the determination of the reference point from three vertices. Thus, the probability of selection of n correct corresponding points is

$$p = \frac{n}{3!n \times n}. \tag{39}$$

For accurate detection of the correspondence, the expected number of iterations N should satisfy the relation

$$pN = n\gamma, \tag{40}$$

because there are n correct correspondences and a threshold γ is set for the maximum in the voting process. This implies that the number of iterations for a polygon is

$$N = 3!n^2\gamma. \tag{41}$$

6. Experimetal Results

In this section we show the results of checking for planarity, checking for planarity of motion and planar motion estimation. All experiments used synthetic data. The algorithm to check planarity computes the parameters which define a plane, and if a unique maximum is detected for these four parameters, then we can conclude that the points form a planar shape. We tested this algorithm using the vertices of the polyhedron shown in Fig. 4, and the coordinates of the vertices are shown in Table 1. We considered two examples, the first one using all polyhedron vertices. In this case the result of cheching for planarity is negative. The second example is using only the vertices of one face (abcde) for both image frames, and the results of the plane parameters are shown in Table 2, which indicate that the vertices lie on the same plane.

To check simultaneously whether the shape and the motion are planar, we performed an experiment using the planar shape shown in Fig. 5, for which the

z axis coordinates are set to unity. The rotation is assumed to be about the z axis, and the translations are on x and y axes respectively. We obtained the same parameters for both frames, which implies that the motion and shape are planar as shown in Table 3. To test the planar motion estimation algorithm we also use the planar shape shown in Fig. 5, the vertices of which are shown in Table 4. The result for the rotation matrix is shown in Table 5, which approximate the true rotation of 45° applied to this planar shape . The results of the translation are shown in Table 6. The number of iterations and the execution time for the computation are shown in Table 7. For the case in which a shape presents some symmetry the solution is up to sign.

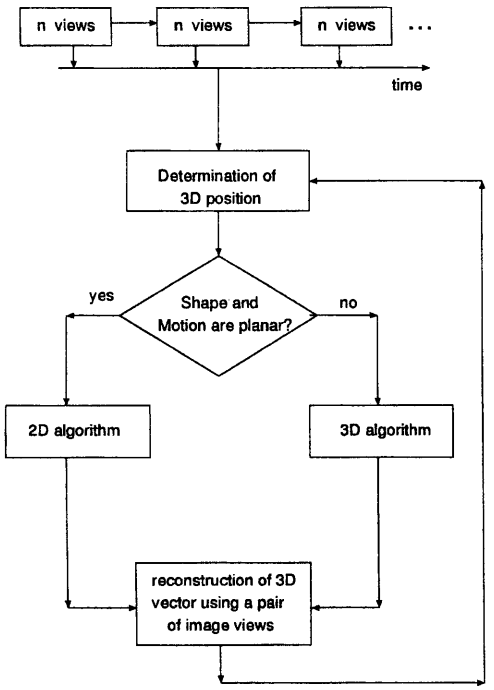

Figure 1: *Motion analysis methodology*

7. Conclusions

We have presented a randomized approach to check for planarity and to check for planarity and planar motion simultaneously and a randomized algorithm for the estimation of motion parameters for a planar shape using the concept of congruence checking. The algorithm for motion estimation for planar shape is applicable to partially occluded objects and it is not necessary to know *a priori* the point cor-

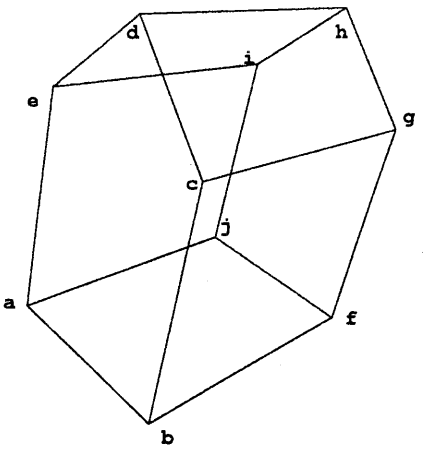

Figure 2: Polyhedron

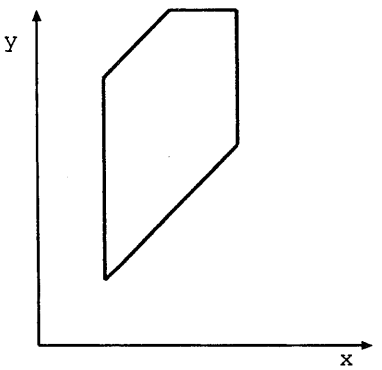

Figure 3: Polygon

Table 1: *Object vertices*

vertices	frame 1			frame 2		
	x	y	z	x	y	z
a	0	0	0	3.0000	5.0000	7.0000
b	4	0	0	3.5309	6.0420	10.8252
c	4	0	5	-0.9240	8.3119	10.8252
d	2	0	7	-2.9715	8.6989	8.9126
e	0	0	5	-1.4550	7.2699	7.0000
f	4	6	0	6.1358	11.1544	9.0709
g	4	6	5	1.6808	13.4244	9.0709
h	2	6	7	-0.3666	13.8113	7.1583
i	0	6	5	1.1498	12.3823	5.2457
j	0	6	0	5.6049	10.1124	5.2457

Table 2: *Plane parameters, face abcde*

	frame 1	frame 2
α	0.000000	-0.118762
β	1.000000	-0.233083
γ	0.000000	0.079978
δ	0.000000	0.079978

Table 3: *Planar shape vertices*

frame 1			frame 2		
x	y	z	x	y	z
2	2	1	3.000000	7.828427	1.000000
8	8	1	3.000000	16.313708	1.000000
8	12	1	0.171573	19.142135	1.000000
6	12	1	-1.242641	17.727922	1.000000
2	10	1	-2.656854	13.485281	1.000000

Table 4: *Planar motion - planarity detection*

	parameters			
	α	β	γ	δ
frame 1	0.0	0.0	-0.707107	0.707107
frame 2	0.0	0.0	-0.707107	0.707107

Table 5: *Rotation matrix*

$\cos^{-1}$	$\sin^{-1}$	$-\sin^{-1}$	$\cos^{-1}$
0.707107	0.707107	-0.707107	0.707107
45.000°	45.000°	45.000°	45.000°

Table 6: *Translation estimation*

	translation	
	x	y
Actual	3.0	5.0
	3.633	5.013

Table 7: *Iterations and execution time*

Iter. R	Iter. t	Time (sec.)
160	750	2.4

respondences. Our algorithm for planar motion estimation is a non-model-based algorithm in which the object features are extracted simultaneously from each image frame to determine the motion parameters directly. The parameters are used to index a hash table where the voting process is executed until a maximum is detected for each motion parameters.

Furthermore, we analyzed the errors of our randomized method for motion detection for two-dimensional space. In conclusion, if we assume that the errors are mainly due to the image digitalization, we obtained the following results.

1. By increasing the resolution of the image of the object, the solutions converge to the correct solutions.

2. The resolution of the accumulator space depends linearly on the object digitalization errors. If an object is small compared to the region of interest, the resolution of the rotation parameters must be small. The resolution of the translation does not depend on the size of objects.

3. Our constraint for checking for vector matching is stable and the threshold does not depend on the resolution.

A. Proof of Theorem 1

For a triplet of vectors α, β, and γ in $\mathbf{R}^3$, we assume that there are errors in the estimation of these vectors, thus

$$\alpha' = \alpha + \varepsilon_\alpha \beta' = \beta + \varepsilon_\beta \gamma' = \gamma + \varepsilon_\gamma, \qquad (42)$$

such that

$$\frac{|\varepsilon_\alpha|}{|\alpha|} \leq \omega, \ \frac{|\varepsilon_\beta|}{|\beta|} \leq \omega, \ \frac{|\varepsilon_\gamma|}{|\gamma|} \leq \omega, \qquad (43)$$

and $|\varepsilon_\alpha| \ll \upsilon \ |\varepsilon_\beta| \ll \upsilon \ |\varepsilon_\gamma| \ll \upsilon$ where υ and ω are constants such that $\upsilon \ll 1$ and $\omega \ll 1$.

Neglecting the terms υ^n for $n \geq 2$, we obtain the relation

$$|\alpha'\beta'\gamma'| = |\alpha\beta\gamma| + \frac{\mu_1}{3}|\varepsilon_\alpha||\beta||\gamma| + \frac{\mu_2}{3}|\alpha||\varepsilon_\beta||\gamma| + \frac{\mu_3}{3}|\alpha||\beta||\varepsilon_\gamma|, \qquad (44)$$

where $|\mu_i| < 3$ for i=1,2,3. Moreover, for a constant $|\kappa| \leq 1$, the relation

$$|\boldsymbol{\alpha}'\boldsymbol{\beta}'\boldsymbol{\gamma}'| = \kappa|\boldsymbol{\alpha}||\boldsymbol{\beta}||\boldsymbol{\gamma}| \tag{45}$$

implies that

$$|\boldsymbol{\alpha}'\boldsymbol{\beta}'\boldsymbol{\gamma}'| = |\boldsymbol{\alpha}\boldsymbol{\beta}\boldsymbol{\gamma}| + |\boldsymbol{\alpha}\boldsymbol{\beta}\boldsymbol{\gamma}|(\frac{\mu_1\omega}{3\kappa_1} + \frac{\mu_2\omega}{3\kappa_2} + \frac{\mu_3\omega}{3\kappa_3}) \tag{46}$$

since $|\boldsymbol{\varepsilon}_\alpha| \ll |\boldsymbol{\alpha}|$, $|\boldsymbol{\varepsilon}_\beta| \ll |\boldsymbol{\beta}|$, $|\boldsymbol{\varepsilon}_\gamma| \ll |\boldsymbol{\gamma}|$, and $\kappa_i \cong 1$. Therefore, we obtain the relation

$$|\boldsymbol{\alpha}'\boldsymbol{\beta}'\boldsymbol{\gamma}'| = |\boldsymbol{\alpha}\,\boldsymbol{\beta}\,\boldsymbol{\gamma}|(1 + \frac{\omega}{\kappa}). \tag{47}$$

Furthermore,

$$\frac{\omega}{\kappa} = \frac{|\boldsymbol{\varepsilon}_\alpha||\boldsymbol{\beta}||\boldsymbol{\gamma}|}{|\boldsymbol{\alpha}\boldsymbol{\beta}\boldsymbol{\gamma}|} \cong \frac{|\boldsymbol{\varepsilon}_\alpha\boldsymbol{\beta}\boldsymbol{\gamma}|}{|\boldsymbol{\alpha}\boldsymbol{\beta}\boldsymbol{\gamma}|}, \tag{48}$$

indicates that ω/κ is the ratio of the volume of the parallelotope spanned by $\boldsymbol{\varepsilon}_\alpha$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$, and $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and $\boldsymbol{\gamma}$. This implies that $\omega/\kappa \ll 1$ since $|\boldsymbol{\varepsilon}_\alpha| \ll |\boldsymbol{\alpha}|$. Therefore, we obtain

$$a' = a(1 + \frac{\omega}{\kappa}) \quad b' = b(1 + \frac{\omega}{\kappa})$$

$$c' = c(1 + \frac{\omega}{\kappa}) \quad d' = d(1 + \frac{\omega}{\kappa}). \tag{49}$$

Thus, setting $\kappa = \min(\kappa_1, \kappa_2, \kappa_3, \kappa_4)$, from Eq. (4) we obtain

$$\begin{aligned} \alpha' &\cong \frac{a(1 + \frac{\omega}{\kappa})}{\sqrt{(a^2 + b^2 + c^2 + d^2)(1 + \frac{\omega}{\kappa})}} \\ &= \alpha(1 - \frac{\omega^2}{\kappa^2}), \end{aligned} \tag{50}$$

we also obtain the same relations for β, γ, and δ,

$$\beta' = \beta(1 - \frac{\omega^2}{\kappa^2}) \;\; \gamma' = \gamma(1 - \frac{\omega^2}{\kappa^2}) \;\; \delta' = \delta(1 - \frac{\omega^2}{\kappa^2}). \tag{51}$$

Hence, we have

$$\begin{aligned} &(\alpha - \alpha')^2 + (\beta - \beta')^2 \\ &+ \;\; (\gamma - \gamma')^2 + (\delta - \delta')^2 = \frac{\omega^4}{\kappa^4}. \end{aligned} \tag{52}$$

References

1 Carlsson S., Projectively invariant descomposition and recognition of planar shapes, *International Journal of Computer Vision*, **17**, pp. 193-209, 1996.

2 Wong, K.C., Kittler, J., Recognising polyhedral objects from a single perpective view, in *Proceedings of British Machine Vision Conferences*, Leeds, pp. 590-599, 1992.

3 Kanatani, K., Tracing planar surface motion from a projection without knowing the correspondence, *Computer Vision, Graphics and Image Processing*, **29**, pp. 1-12, 1985.

4 Aloimonos, J., Hervé, J., Correspondenceless stereo and motion: Planar surfaces, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, pp. 504-510, 1990.

5 Imiya, A., Fermin, I., Motion analysis by random sampling and voting, in preparation.

6 Blostein, S., Huang, T.S., Error analysis in stereo determination of 3-D point positions, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**, pp. 752-765, 1987.

7 Rodriguez, I.J., Aggarwal, J.K., Stochastic analysis of stereo quantization error, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **12**, pp. 467-470, 1990.