

# Using moments to acquire the motion parameters of a deformable object without correspondences

Soo-Chang Pei and Lin-Gwo Liou

---

In the topic of motion analysis in image understanding, we usually need to find the motion of a particular object. However, frequently the correspondence problem, which is well known to be difficult, must first be solved. If the correspondence problem can be avoided in motion analysis, it will greatly reduce the difficulty of solving motion parameters. Using the concept of object normalization described in this paper, we can determine the motion parameters without solving the correspondence problem. The object of interest is not necessarily assumed rigid here. It can undergo a uniformly, linearly deformable motion (affine transformation), including translation, rotation and skewing. Under the assumption of this motion model, a closed-form solution of the desired motion parameters can be obtained efficiently by our proposed method. In some special cases where the closed-form solution cannot be determined uniquely, a modified method is also developed to solve them. A theoretical error analysis is presented to show for what kinds of object it is inherently hard to acquire motion parameters.

**Keywords:** moment, shape normalization, affine transformation, rigid object, uniformly-deformable

---

Before introducing this topic, we have to explain the character of the so-called *interested* object in our paper. In 2D space, the interested object can be a continuous shape or just a collection of points. It can move in 2D space in a rigid or deformable way. In 3D space, the interested object can be a continuous solid volume, surface or just a collection of points. Similarly, it can also move in 3D space in a rigid or deformable way. The

---

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, Republic of China  
Paper received: 6 November 1993; revised paper received: 12 April 1994

motion model used here is constrained to the linearly-deformable one (affine transformation). The details of this model will be clearly defined in later sections.

Previously, it has usually been necessary to solve the correspondence problem before estimating the motion parameter of an interested object. However, the correspondence problem is quite hard to solve, therefore many researchers try to find some *correspondenceless* methods. Until now, there have been some good results published: the concept of moments (or tensor) is one of the best choices. In fact, in the topic of pattern recognition, moments have been adopted extensively in areas such as moment invariants<sup>1,7</sup>, orientation determination for the planar patch<sup>8,11</sup>, shape normalization<sup>12</sup>, object detection<sup>13</sup>, etc. In these applications, two concepts are especially appealing to us:

1. Affine-invariants defined by moments, which says that two 2D shapes different by an arbitrary affine transformation have some *unchangeable* common quantities which can be utilized for recognition.
2. Shape normalization, which says that two 2D shapes different by an arbitrary affine transformation can be both transformed to a normalized shape.

We feel that these ideas can also be utilized for correspondenceless motion estimation, especially for the affine motion model.

Related research on correspondenceless motion estimation has been published. For example, Chaudhuri and Chatterjee<sup>14</sup> proposed a method which can solve the generalized motion parameters of a uniformly-deformable 3D object with the knowledge of several subsets of corresponding points. Although this method solves some difficulties in finding correspondences, it is

not convenient to use because we still need to know several subsets of correspondences. Faber and Stokely<sup>15</sup> solve a similar problem about medical application by moments. Pei *et al.*<sup>16,17</sup> propose several correspondenceless method to estimate the 3D motion of a planar patch in 3D space. The heart of these methods is the calculation of moments and the linear-deformation motion models.

The very early ideas of our methods come from 2D shape normalization<sup>12</sup>. Leu began his derivations from 2D shapes and used them in 2D shape recognition. In this paper, we extend his concept to correspondenceless motion estimation of linearly-deformable 2D shapes and 3D objects. The methods proposed here are easy and computationally efficient. To reduce the cost of moment calculation, Green's theorem<sup>18</sup> or another similar method<sup>19</sup> can be adopted in our calculations. Finally, to complete the theory of motion estimation, several difficult cases such as ambiguities and degeneracy are discussed. A theoretical error analysis of the motion estimation is also proposed.

Our paper is organized as follows. The next section discusses correspondenceless motion estimation in the 2D case. Correspondenceless motion estimation in the 3D case is then examined, and a theoretical error analysis carried out. Several simulation experiments are outlined to prove the validity of the methods described, and conclusions are finally drawn.

**CORRESPONDENCELESS MOTION ESTIMATION IN 2D SPACE**

In this section, we derive a method for correspondenceless motion estimation in 2D space. The object of interest considered here can be a 2D continuous shape or just a collection of several discrete points. Without loss of generality, only equations about a continuous 2D shape are listed here. The derivations about the discrete-point structure are very similar. All we have to do is change the definitions of continuous moments to discrete moments.

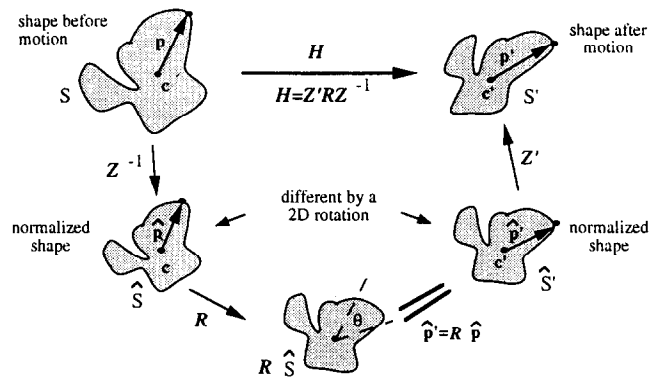
**Problem formulation**

Consider a 2D shape  $S$ . It moves to another new position  $S'$  according to the following formula (see Figure 1):

$$\mathbf{p}' \equiv \begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{H} \begin{bmatrix} x \\ y \end{bmatrix} + \mathbf{t} \equiv \mathbf{H}\mathbf{p} + \mathbf{t} \tag{1}$$

where  $\mathbf{H}$  is a  $2 \times 2$  transform matrix, and  $\mathbf{p}$  and  $\mathbf{p}'$  separately stand for the corresponding points before and after motion. Equation (1) defines the affine motion model which includes translating, scaling, rotating and skewing in 2D space.

All we know as input data are two 2D shapes,  $S$  and  $S'$ . However, the one-to-one point correspondences



**Figure 1** Normalization process of two 2D shapes that are different by an affine transformation

between them are entirely unknown. We also assume that both these shapes are complete, and there are no missing parts. Under the above conditions, our final goal is to determine the unknown transform matrix  $\mathbf{H}$  and translation vector  $\mathbf{t}$ .

Before describing our main algorithm, some points must be considered:

1. Camera effects, such as perspective projection or distortion, are not considered here, i.e. we do not consider how the 2D input data is obtained.
2. Only the case that  $\det(\mathbf{H}) > 0$  is considered here. The problem of mirror projection,  $\det(\mathbf{H}) < 0$ , can be solved by a slight modification to the equations, and they are not discussed in this paper for simplicity.

**Main algorithm**

Because the 2D shape  $S$  undergoes an affine motion, the mass centre  $\mathbf{c}$  of  $S$  must correspond to the mass centre  $\mathbf{c}'$  of the new shape  $S'$ . Here the mass centres are defined as:

$$\mathbf{c} = \frac{1}{|S|} \int \int_S \mathbf{p} dS; \quad \mathbf{c}' = \frac{1}{|S'|} \int \int_{S'} \mathbf{p}' dS' \tag{2}$$

Therefore, without loss of generality, the points on the object can be represented by an object-centred coordinate system. To simplify the notation, we assume that the position vectors  $\mathbf{p}$  and  $\mathbf{p}'$  have been separately expressed by their own object-centred coordinate systems. So, we have  $\mathbf{c} = 0$ ,  $\mathbf{c}' = 0$  and  $\mathbf{t} = 0$ . The transform described in equation (1) can now be simplified to:

$$\mathbf{p}' \equiv \mathbf{H}\mathbf{p} \tag{3}$$

The  $2 \times 2$  dispersion matrices  $\mathbf{M}$  and  $\mathbf{M}'$  of shapes  $S$  and  $S'$  can be separately defined as:

$$\mathbf{M} \equiv \frac{1}{|S|} \int \int_S \mathbf{p}\mathbf{p}^T dS; \quad \mathbf{M}' \equiv \frac{1}{|S'|} \int \int_{S'} \mathbf{p}'\mathbf{p}'^T dS' \tag{4}$$

where  $|S|$  and  $|S'|$  are separately the areas of  $S$  and  $S'$ , and the superscript  $T$  denotes the transpose operation.

Notice the elements of the dispersion matrices; they are just equal to the second-order moments of the shapes.

After substituting equation (3) into equation (4), we have:

$$\mathbf{M}' = \mathbf{H}\mathbf{M}\mathbf{H}^T \quad (5)$$

The two dispersion matrices  $\mathbf{M}$  and  $\mathbf{M}'$  can be separately decomposed by similarity transformation, defined as follows:

$$\mathbf{M} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^T = (\mathbf{Q}\mathbf{\Lambda}_{1/2})\mathbf{I}(\mathbf{Q}\mathbf{\Lambda}_{1/2})^T \equiv \mathbf{Z}\mathbf{I}\mathbf{Z}^T \quad (6)$$

$$\mathbf{M}' = \mathbf{Q}'\mathbf{\Lambda}'\mathbf{Q}'^T = (\mathbf{Q}'\mathbf{\Lambda}'_{1/2})\mathbf{I}(\mathbf{Q}'\mathbf{\Lambda}'_{1/2})^T \equiv \mathbf{Z}'\mathbf{I}\mathbf{Z}'^T \quad (7)$$

Here  $\mathbf{Q}$  and  $\mathbf{Q}'$  are orthogonal matrices and their determinants must be equal to +1 (avoiding the mirror projection).  $\mathbf{\Lambda}$  and  $\mathbf{\Lambda}'$  are two diagonal matrices with non-negative elements.  $\mathbf{\Lambda} = \mathbf{\Lambda}_{1/2}\mathbf{\Lambda}_{1/2}$  and  $\mathbf{\Lambda}' = \mathbf{\Lambda}'_{1/2}\mathbf{\Lambda}'_{1/2}$ .  $\mathbf{I}$  is a  $3 \times 3$  identity matrix.

After substituting equations (6) and (7) into equation (5), we have:

$$\mathbf{Z}'\mathbf{Z}'^T = \mathbf{H}\mathbf{Z}\mathbf{Z}^T\mathbf{H}^T = (\mathbf{H}\mathbf{Z}\mathbf{R}^T)(\mathbf{H}\mathbf{Z}\mathbf{R}^T)^T \quad (8)$$

where  $\mathbf{R}$  is a  $2 \times 2$  orthogonal matrix. It is easy to observe that:

$$\mathbf{H} = \mathbf{Z}'\mathbf{R}\mathbf{Z}^{-1} \quad (9)$$

Equation (9) is the most important equation in this algorithm. In fact, equation (9) can be more clearly interpreted by the concept of normalization. Let us again see *Figure 1*. If we change the two original shapes ( $\hat{S}$  and  $\hat{S}'$ ) to new shapes ( $\hat{S}$  and  $\hat{S}'$ ) by the following transforms:

$$\hat{\mathbf{p}} \equiv \mathbf{Z}^{-1}\mathbf{p}; \quad \hat{\mathbf{p}}' \equiv \mathbf{Z}'^{-1}\mathbf{p}' \quad (10)$$

it is easy to prove that the dispersion matrices of the two new shapes  $\hat{S}$  and  $\hat{S}'$  are both equal to an identity matrix. We may call  $\hat{S}$  and  $\hat{S}'$  the *normalized shapes*.

From the description in Leu<sup>12</sup>, we know that the two normalized objects are only different by a 2D rotation matrix  $\mathbf{R}$ , i.e.:

$$\hat{\mathbf{p}}' = \mathbf{R}\hat{\mathbf{p}} \quad (11)$$

Combining equations (3), (10) and (11), we have:

$$\mathbf{p}' = (\mathbf{Z}'\mathbf{R}\mathbf{Z}^{-1})\mathbf{p} = \mathbf{H}\mathbf{p} \quad (12)$$

Comparing equation (12) with equation (9), we find that both of them represent the transform matrix  $\mathbf{H}$  by the same form  $\mathbf{Z}'\mathbf{R}\mathbf{Z}^{-1}$ . On the other hand, the physical meaning of the orthonormal matrix  $\mathbf{R}$  is now clearly defined. Matrix  $\mathbf{R}$  is just the rotation difference between the two normalized objects. Because we do not consider the case that  $\det(\mathbf{H}) < 0$  (the mirror projection), the value  $\det(\mathbf{R})$  must be equal to 1.

Once the rotation matrix  $\mathbf{R}$  is determined, the transform matrix  $\mathbf{H}$  will be obtained easily by equation (9), and the whole problem is completely solved. The methods of determining  $\mathbf{R}$  will be described later.

The above derivations can be extended directly to the

discrete-point 2D structures by slightly modifying the definitions of dispersion matrices  $\mathbf{M}$  and  $\mathbf{M}'$ :

$$\mathbf{M} \equiv \frac{1}{N} \sum_{i=1}^N \mathbf{p}_i \mathbf{p}_i^T; \quad \mathbf{M}' \equiv \frac{1}{N} \sum_{i=1}^N \mathbf{p}'_i \mathbf{p}'_i^T \quad (13)$$

The key equation (9) is then usable.

## Determine the 2D rotation matrix

In this subsection, we consider two main approaches: the weighting method and the modified method. Basically, these two methods are very similar. The differences between them come from the different utilization of the normalized shapes,  $\hat{S}$  and  $\hat{S}'$ .

### Weighting method

Consider a single-valued weighting function  $g(r)$ . From equation (11), we can immediately obtain the following equation:

$$g(r')\hat{\mathbf{p}}' = \mathbf{R}(g(r)\hat{\mathbf{p}}) \quad (14)$$

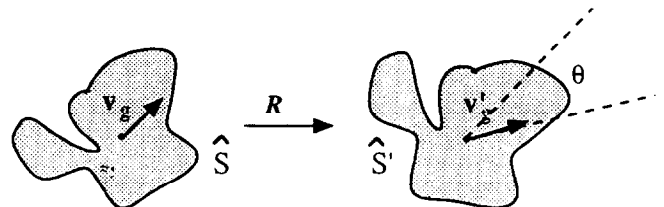
where  $r \equiv \|\hat{\mathbf{p}}\| = \|\hat{\mathbf{p}}'\| \equiv r'$  and  $g(r) = g(r')$ .

If we separately integrate both sides of equation (14) within the regions of  $\hat{S}$  and  $\hat{S}'$ , we will have:

$$\begin{aligned} \mathbf{v}'_g &\equiv \frac{1}{|\hat{S}'|} \int_{\hat{S}'} g(r')\hat{\mathbf{p}}' d\hat{S}' = \mathbf{R} \left( \frac{1}{|\hat{S}|} \int_{\hat{S}} g(r)\hat{\mathbf{p}} d\hat{S} \right) \\ &\equiv \mathbf{R}\mathbf{v}_g \end{aligned} \quad (15)$$

To solve the 2D rotation  $\mathbf{R}$ , at least one weighting function  $g$  is needed. Be sure that the function  $g$  will not make the weighted summation vector  $\mathbf{v}_g$  vanish. From this nonzero vector  $\mathbf{v}_g$ , we can directly estimate the angle  $\theta$  representing the rotation matrix  $\mathbf{R}$ . *Figure 2* shows these ideas.

Is there any constraint that  $g(r)$  must satisfy? Here is a rule: 'Theoretically, any single-value bounded function  $g(r)$  that will not let the weighted summation vector  $\mathbf{v}_g$  become zero will work'. For example,  $g(r) = r^k$ ,  $\sin(r)$  or  $e^r$  are valid candidates; but  $g(r) = \text{constant}$  or  $1/r$  are inappropriate forms. Most of the time, we adopt simpler forms such as  $g(r) = r^2$  or  $g(r) = r^{1/2}$ . Readers may have found that equation (15) is in fact a relationship among higher (or fractional) order moments. This result seems reasonable when compared with published papers about moment invariants.



**Figure 2** In the weighting method, the difference angle  $\theta$  between the two weighted-integration vectors  $\mathbf{v}_g$  and  $\mathbf{v}'_g$  can be used to estimate the rotation matrix  $\mathbf{R}$

When the object of interest is a discrete-point 2D structure, the weighted-integration in equation (15) should be changed to the weighted-summation, defined as:

$$\mathbf{v}'_g \equiv \frac{1}{N} \sum_{i=1}^N g(r'_i) \hat{\mathbf{p}}'_i = \mathbf{R} \left( \frac{1}{N} \sum_{i=1}^N g(r_i) \hat{\mathbf{p}}_i \right) \equiv \mathbf{R} \mathbf{v}_g \quad (16)$$

**Modified method**

The method described here still uses the property of normalized objects (equation (11)); however, different to the weighting method, we no longer calculate any weighted-integration (or summation) vectors of normalized objects. On the contrary, we try to find the best matching between the two normalized shapes  $\hat{S}$  and  $\hat{S}'$ , then the best rotation matrix  $\mathbf{R}$  is substituted into equation (9) to determine  $\mathbf{H}$ . The matrix  $\mathbf{R}$  estimated here is usually more reliable than that done by the weighting method, because the reliability of the weighting method depends upon the amplitude of  $\mathbf{v}_g$  and any position errors will directly affect the values of weighting function  $g(r)$ . Therefore, we may call this new method the *modified* method. Experiments will show its excellence in error performance.

An initial best matching guess can be obtained using several simple methods (e.g. the weighting method). However, the simplest method is to detect the furthest (or the nearest) points of  $\hat{S}$  and  $\hat{S}'$  and calculate their angle difference (Figure 3). Of course, we have to test several initial guesses if the furthest point of a normalized is not unique.

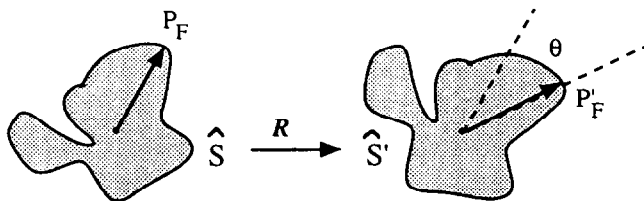
If the object of interest is a discrete-point 2D structure, it will be better to determine directly the one-to-one point correspondences between its normalized objects, because the number of corresponding points is finite, not like the infinite number in a continuous 2D shape. Once the correct one-to-one point correspondences are found, it will be straightforward to determine  $\mathbf{H}$  by solving:

$$[\mathbf{p}'_1 | \dots | \mathbf{p}'_N] = \mathbf{H} [\mathbf{p}_1 | \dots | \mathbf{p}_N] \quad (17)$$

**Discussion**

In this subsection, we discuss the benefits and shortages of the two methods described above.

In the weighting method, the reliability of the estimate of 2D rotation  $\mathbf{R}$  depends upon the amplitude



**Figure 3** In the modified method, the best rotation  $\mathbf{R}$  can be estimated by maximizing the matching ratio between  $\mathbf{R} \hat{S}$  and  $\hat{S}'$ . We may use the furthest points  $P_F$  and  $P'_F$  of the two normalized shapes to guess an initial solution

of vector  $\mathbf{v}_g$ . It no longer works if its amplitude is zero. For example, vector  $\mathbf{v}_g$  of a centre-symmetrical normalized 2D shape is always zero, no matter what kind of function  $g$  is used. This makes the weighting method break down. However, because the modified method does not use weighted integration at all, the singular cases breaking the weighting method do not cause any problem when using the modified method. Of course, these improvements are only obtained by paying the price of greater computational cost.

If the normalized shape  $\hat{S}$  (or  $\hat{S}'$ ) is fold-symmetric (Figure 4), multiple solutions of  $\mathbf{R}$  (and then matrix  $\mathbf{H}$ ) should be discussed. The weighting method usually fails in these cases, but the modified method does not.

**CORRESPONDENCELESS MOTION ESTIMATION IN 3D SPACE**

In this section, we derive a method for correspondenceless motion estimation in 3D space. The object of interest considered here can be a 3D solid volume or just a collection of several discrete points. Without loss of generality, only the equations about a discrete-point 3D structure are listed here. The derivations about 3D solid volume are very similar; we just need to change the definitions of discrete moments to continuous moments.

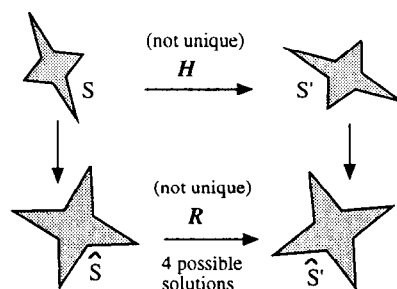
**Problem formulation**

Consider a discrete-point 3D structure  $S$ . It moves to another new position  $S'$  according to the formula:

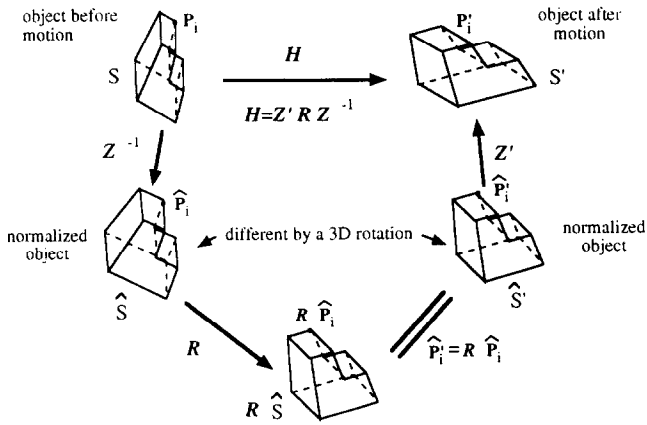
$$\mathbf{P}'_i \equiv \begin{bmatrix} X'_i \\ Y'_i \\ Z'_i \end{bmatrix} = \mathbf{H} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} + \mathbf{T} \equiv \mathbf{H} \mathbf{P}_i + \mathbf{T} \quad (18)$$

where  $\mathbf{H}$  is a  $3 \times 3$  transform matrix, and  $\mathbf{P}_i$  and  $\mathbf{P}'_i$  (for  $i = 1, \dots, N$ ) separately stand for the corresponding points on  $S$  and  $S'$ . Equation (18) defines the affine motion model, which includes translating, scaling, rotating and skewing in 3D space (Figure 5).

All we know as input data are the 3D coordinates of  $\mathbf{P}_i$ s and  $\mathbf{P}'_i$ s, which have been recognized as the points on the same moving object. However, the one-to-one point correspondences between  $S$  and  $S'$  are unknown.



**Figure 4** One of the ambiguity cases. If the normalized shapes are rotationally-symmetric, there may exist several possible solutions for  $\mathbf{H}$



**Figure 5** Normalization process of two 3D objects that are different by an affine transformation

The problem of missing points is not considered here. Under the above conditions, our final goal is to determine the unknown transform matrix  $H$  and translation vector  $T$ .

Before beginning our main derivations, some points must be considered:

1. We do not consider how the 3D point data is obtained (range camera or computer-aided tomography).
2. Only the case where  $\det(H) > 0$  is considered here. Similar to the 2D case, the problem of mirror projection ( $\det(H) < 0$ ) can be solved easily by a slight modification to the equations, so they are not discussed in this paper.

## Main algorithm

Without loss of generalization, we may assume that the position vectors  $\mathbf{P}_i$ s and  $\mathbf{P}'_i$ s have been expressed by their own object-centred coordinate system, and equation (18) can now be simplified to:

$$\mathbf{P}'_i \equiv H\mathbf{P}_i \quad \text{for } i = 1, \dots, N \quad (19)$$

The  $3 \times 3$  dispersion matrices,  $M$  and  $M'$ , of  $S$  and  $S'$  can be separately defined as:

$$M \equiv \frac{1}{N} \sum_{i=1}^N \mathbf{P}_i \mathbf{P}_i^T; \quad M' \equiv \frac{1}{N} \sum_{i=1}^N \mathbf{P}'_i \mathbf{P}'_i^T \quad (20)$$

where the superscript  $T$  denotes the transpose operation.

The two dispersion matrices  $M$  and  $M'$  can be decomposed separately by similarity transformation, defined as follows:

$$M = \mathbf{Q}\Lambda\mathbf{Q}^T = (\mathbf{Q}\Lambda_{1/2})\mathbf{I}(\mathbf{Q}\Lambda_{1/2})^T \equiv \mathbf{Z}\mathbf{I}\mathbf{Z}^T \quad (21)$$

$$M' = \mathbf{Q}'\Lambda'\mathbf{Q}'^T = (\mathbf{Q}'\Lambda'_{1/2})\mathbf{I}(\mathbf{Q}'\Lambda'_{1/2})^T \equiv \mathbf{Z}'\mathbf{I}\mathbf{Z}'^T \quad (22)$$

Here  $\mathbf{Q}$  and  $\mathbf{Q}'$  are orthogonal matrices and their determinants must be equal to +1 (preventing mirror projection).  $\Lambda$  and  $\Lambda'$  are two diagonal matrices with nonnegative elements, and  $\mathbf{I}$  is a  $3 \times 3$  identity matrix.

Readers must have found great similarity between the 2D and 3D cases. Therefore, the 3D objects,  $S$  and  $S'$ , can also be normalized to two new objects,  $\hat{S}$  and  $\hat{S}'$  by defining:

$$\hat{\mathbf{P}}_i \equiv \mathbf{Z}^{-1}\mathbf{P}_i; \quad \hat{\mathbf{P}}'_i \equiv \mathbf{Z}'^{-1}\mathbf{P}'_i \quad (23)$$

Here the dispersion matrices  $\hat{M}$  and  $\hat{M}'$  of the two normalized objects  $\hat{S}$  and  $\hat{S}'$  must be equal to a  $3 \times 3$  identity matrix  $\mathbf{I}$  (Figure 5).

From the description given by Leu<sup>12</sup>, we know that  $\hat{S}$  and  $\hat{S}'$  are different by a 3D rotation matrix  $\mathbf{R}$ , i.e.:

$$\hat{\mathbf{P}}'_i = \mathbf{R}\hat{\mathbf{P}}_i \quad (24)$$

Combining equations (19), (23) and (24), we also derive the same key equation as that in equation (9):

$$H = \mathbf{Z}'\mathbf{R}\mathbf{Z}^{-1} \quad (25)$$

The value  $\det(\mathbf{R})$  must be equal to +1 because  $\det(H)$  is positive.

Once the rotation matrix  $\mathbf{R}$  has been determined, the transform matrix  $H$  will be obtained easily by equation (25), and the whole problem is solved. The methods of determining  $\mathbf{R}$  are described later.

The above derivations can easily be extended to a continuous 3D solid volume by slightly modifying the definitions of dispersion matrices  $M$  and  $M'$ :

$$M \equiv \frac{1}{|S|} \int \int \int_S \mathbf{P}\mathbf{P}^T dS; \quad M' \equiv \int \int \int_{S'} \mathbf{P}'\mathbf{P}'^T dS' \quad (26)$$

Here,  $|S|$  and  $|S'|$  are separately the volumes of the objects  $S$  and  $S'$ . After this modification, equation (25) can also be used to determine the transform matrix  $H$ .

## Determine the 3D rotation matrix

In this subsection, we also consider two main approaches, the weighting method and the modified method, as in the 2D case.

### Weighting method

Consider a single-valued weighting function  $g(r)$ . Here the function  $g$  should satisfy the thumb's rule described above. From equation (24), we can immediately obtain the equation:

$$g(r'_i)\hat{\mathbf{P}}'_i = \mathbf{R}(g(r_i)\hat{\mathbf{P}}_i) \quad (27)$$

where  $r_i \equiv \|\hat{\mathbf{P}}_i\| = \|\hat{\mathbf{P}}'_i\| \equiv r'_i$  and  $g(r'_i) = g(r_i)$  for  $i = 1, \dots, N$ .

If we sum both sides of equation (27) from  $i = 1, \dots, N$ , we have:

$$\mathbf{V}'_g \equiv \frac{1}{N} \sum_{i=1}^N \{g(r'_i)\hat{\mathbf{P}}'_i\} = \mathbf{R} \left( \frac{1}{N} \sum_{i=1}^N \{g(r_i)\hat{\mathbf{P}}_i\} \right) \equiv \mathbf{R}\mathbf{V}_g \quad (28)$$

To solve the 3D rotation  $\mathbf{R}$ , at least two different functions (e.g.  $g_1$  and  $g_2$ ) must be specified to acquire

two different sets of vectors, say  $\{\mathbf{V}_{g1}, \mathbf{V}_{g2}\}$  and  $\{\mathbf{V}'_{g1}, \mathbf{V}'_{g2}\}$ . In each set, we must also make sure that its two element vectors will not be zero or parallel to each other (Figure 6).

If subset correspondences are available, as mentioned by Chaudhuri and Chatterjee<sup>14</sup>, we do not have to sum up all the weighting vectors from index  $1, \dots, N$ . For example, if  $K (> 1)$  subsets of correspondences are available, we just need to sum up the  $g$ -weighted vectors in the same subset. Therefore,  $K$  different pairs of vectors, like  $\{\mathbf{V}_g, \mathbf{V}'_g\}$  can be obtained. So the 3D rotation estimate based on these  $K$  corresponding vectors will be more reliable. It seems that the weighting method has a greater flexibility.

When the object of interest is a continuous 3D volume, the weighted-summation in equation (28) should be changed to the weighted-integration defined as:

$$\begin{aligned} \mathbf{V}'_g &\equiv \frac{1}{|\hat{S}'|} \int \int \int_{\hat{S}'} g(r') \hat{\mathbf{P}}' d\hat{S}' \\ &= \mathbf{R} \left\{ \frac{1}{|\hat{S}|} \int \int \int_{\hat{S}} g(r) \hat{\mathbf{P}} d\hat{S} \right\} \equiv \mathbf{R} \mathbf{V}_g \end{aligned} \quad (29)$$

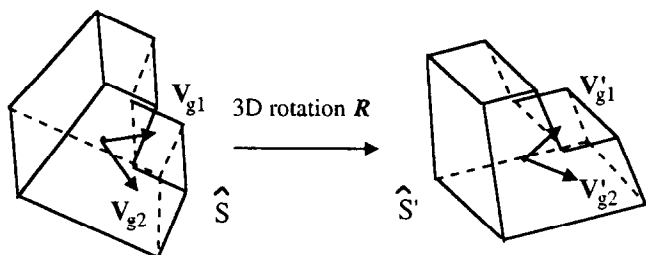
**Modified method**

Because the discrete-point 3D structure has a finite number of one-to-one point correspondences, the modified method described here will focus on finding the correct one-to-one point correspondence between  $\hat{S}$  and  $\hat{S}'$ . Once the correct one-to-one point correspondences have been determined, the transform matrix  $\mathbf{H}$  can be estimated directly by solving:

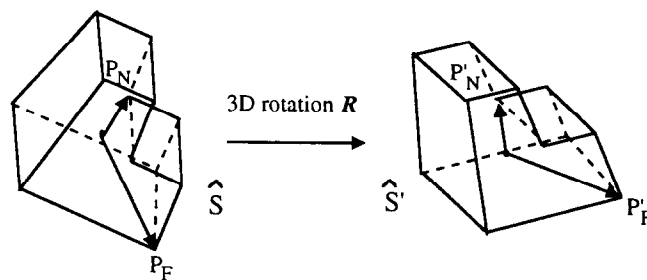
$$[\mathbf{P}'_1 | \dots | \mathbf{P}'_N] = \mathbf{H} [\mathbf{P}_1 | \dots | \mathbf{P}_N] \quad (30)$$

Many methods can serve to provide an initial guess of the correct one-to-one point correspondences. The weighting method described earlier is one of them, but the simplest way is to find the corresponding furthest and nearest points on the two normalized objects  $\hat{S}$  and  $\hat{S}'$ . These points can be separately denoted by  $P_F, P_N, P'_F$  and  $P'_N$  (Figure 7). From these points, we can obtain an estimate of  $\mathbf{R}$ , which helps the later work of finding the correct correspondences. Of course, we have to test several initial guesses if these extreme points are not unique.

When the object of interest is a continuous 3D solid volume, the modified method will focus on finding the



**Figure 6** In the weighting method, two pairs of weighted-summation vectors are used to determine the unknown 3D rotation  $\mathbf{R}$  between  $\hat{S}$  and  $\hat{S}'$



**Figure 7** In the modified method, the best rotation  $\mathbf{R}$  can be estimated by maximizing the matching ratio between  $\mathbf{R} \hat{S}$  and  $\hat{S}'$ . We may use the furthest and nearest points  $\{P_F, P_N, P'_F, P'_N\}$  of the two normalized objects to guess an initial solution

rotation  $\mathbf{R}$ , which causes a best matching between its normalized objects. This is because the number of corresponding points is infinite in this case.

**Discussion**

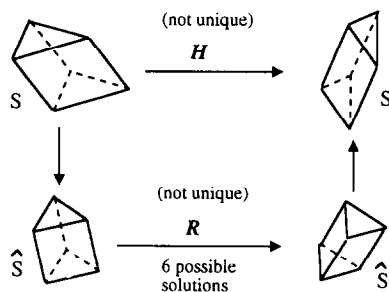
In this subsection, we discuss the benefits and shortcomings of the two methods described above.

The weighting method will fail when (1) any one of  $\mathbf{V}_{g1}$  and  $\mathbf{V}_{g2}$  is zero, or (2)  $\mathbf{V}_{g1}$  is parallel to  $\mathbf{V}_{g2}$ . For example, the weighted-summation vector  $\mathbf{V}_g$  of a centre-symmetrical normalized object is always zero no matter what kind of function  $g$  is used, which makes the weighting method break down. On the other hand, the two functions,  $g_1$  and  $g_2$  should be chosen appropriately for not making the opening angle between  $\mathbf{V}_{g1}$  and  $\mathbf{V}_{g2}$  too small. However, the modified method does solve these singular cases, because it never uses the weighted summation (or integration). Of course, these improvements are obtained only at the cost of more computation.

If the normalized 3D object  $\hat{S}$  and  $\hat{S}'$  have some kind of rotational symmetry (Figure 8), we may have multiple solutions for  $\mathbf{R}$  (then matrix  $\mathbf{H}$ ). In this case, the weighting method usually fails, but the modified method can solve all the possible solutions.

**ERROR ANALYSIS**

In this section, we discuss what factors will affect the error performance of the weighting and modified



**Figure 8** One of the ambiguity cases. If the normalized objects are rotationally-symmetric in 3D space, there may exist several possible solutions for  $\mathbf{H}$

methods. Because of the great similarity between the 2D and 3D cases, we only consider error analysis of the 3D case here. If we first assume an ideal case, where the correspondence between  $\mathbf{P}_i$ s and  $\mathbf{P}'_i$ s is known, then we have:

$$[\mathbf{P}'_1 | \cdots | \mathbf{P}'_N] = \mathbf{H}[\mathbf{P}_1 | \cdots | \mathbf{P}_N] \quad (31)$$

If  $N \geq 3$ , the transform matrix  $\mathbf{H}$  can be solved by the equation:

$$\mathbf{H} = \left( \frac{1}{N} \sum_{i=1}^N \mathbf{P}'_i \mathbf{P}'_i^T \right) \left( \frac{1}{N} \sum_{i=1}^N \mathbf{P}_i \mathbf{P}_i^T \right)^{-1} \equiv \mathbf{B} \mathbf{M}^{-1} \quad (32)$$

How will the estimate of  $\mathbf{H}$  be affected if the position vectors  $\mathbf{P}_i$ s and  $\mathbf{P}'_i$ s are perturbed by noise? If the perturbed forms of these position vectors are represented by  $\tilde{\mathbf{P}}_i$ s and  $\tilde{\mathbf{P}}'_i$ s, then we have the perturbed forms of  $\mathbf{B}$ ,  $\mathbf{M}$ ,  $\mathbf{H}$ :  $\tilde{\mathbf{B}} = \mathbf{B} + \Delta\mathbf{B}$ ,  $\tilde{\mathbf{M}} = \mathbf{M} + \Delta\mathbf{M}$ ,  $\tilde{\mathbf{H}} = \mathbf{H} + \Delta\mathbf{H}$  and  $\tilde{\mathbf{H}} = \tilde{\mathbf{B}} \tilde{\mathbf{M}}^{-1}$ . Substituting them into equation (32), we have:

$$\begin{aligned} \Delta\mathbf{H} &\equiv \tilde{\mathbf{H}} - \mathbf{H} \approx -\mathbf{H}(\Delta\mathbf{M})\mathbf{M}^{-1} + (\Delta\mathbf{B})\mathbf{B}^{-1}\mathbf{H} \\ &= -\mathbf{H}(\Delta\mathbf{M})\mathbf{M}^{-1} + (\Delta\mathbf{B})\mathbf{M}^{-1} \end{aligned} \quad (33)$$

where the second-order errors are neglected.

We further define the relative estimation error of  $\mathbf{H}$  as  $\|\Delta\mathbf{H}\|_F / \|\mathbf{H}\|_F$ , where  $\|\cdot\|_F$  is defined as the Frobenius norm. Finally, an upper bound of this relative error can be derived as:

$$\frac{\|\Delta\mathbf{H}\|_F}{\|\mathbf{H}\|_F} \leq \|\Delta\mathbf{M}\|_F \|\mathbf{M}^{-1}\|_F + \|\Delta\mathbf{B}\|_F \|\mathbf{B}^{-1}\|_F \quad (34)$$

From equation (34), it is easy to see that if matrix  $\mathbf{M}$  or  $\mathbf{B}$  has an eigenvalue which is very close to zero, then the final estimated motion matrix will be very error-sensitive. A close-to-zero eigenvalue of  $\mathbf{M}$  means that the object structure is degenerate, such as a planar patch in 3D space or a 1D line in 2D space. When the correct correspondences are available, the error performance only depends upon the structures before and after motion, which also means that the error performance will not be affected, no matter how the correct correspondences are obtained. So we know that the performance of the modified method must be the same as this ideal case.

As for the weighting method, the error sensitivity of the estimated motion matrix  $\mathbf{H}$  is quite different. Let us begin from the error-perturbed form of equation (25), i.e.:

$$\tilde{\mathbf{H}} = \tilde{\mathbf{Z}}' \tilde{\mathbf{R}} \tilde{\mathbf{Z}}^{-1} \quad (35)$$

where  $\tilde{\mathbf{Z}}' = \mathbf{Z}' + \Delta\mathbf{Z}'$ ,  $\tilde{\mathbf{R}} = (\Delta\mathbf{R})\mathbf{R} = (\mathbf{I} + \Delta\mathbf{W})\mathbf{R}$  and  $\tilde{\mathbf{Z}} = \mathbf{Z} + \Delta\mathbf{Z}$ .  $\Omega = (\Omega_x, \Omega_y, \Omega_z)^T$  is the rotation vector representing the  $3 \times 3$  matrix  $\mathbf{R}$ .  $\Delta\mathbf{W}$  is an anti-symmetric matrix defined as:

$$\Delta\mathbf{W} \equiv \begin{bmatrix} 0 & -\Delta\Omega_z & \Delta\Omega_y \\ \Delta\Omega_z & 0 & -\Delta\Omega_x \\ -\Delta\Omega_y & \Delta\Omega_x & 0 \end{bmatrix} \quad (36)$$

After substituting equations (25) and (36) into the definition of  $\Delta\mathbf{H}$ , we have

$$\begin{aligned} \Delta\mathbf{H} &\equiv \tilde{\mathbf{H}} - \mathbf{H} \approx [(\Delta\mathbf{Z}')(\mathbf{Z}')^{-1}]\mathbf{H} + \mathbf{H}[(\Delta\mathbf{Z})(\mathbf{Z})^{-1}] \\ &\quad + \mathbf{Z}'[(\Delta\mathbf{W})\mathbf{R}]\mathbf{Z}^{-1} \end{aligned} \quad (37)$$

Similar to equation (34), we derive an upperbound of the relative error,  $\|\Delta\mathbf{H}\|_F / \|\mathbf{H}\|_F$ , as follows:

$$\begin{aligned} \text{relative error} &\equiv \frac{\|\Delta\mathbf{H}\|_F}{\|\mathbf{H}\|_F} \leq \|\Delta\mathbf{Z}'\|_F \|\mathbf{Z}'^{-1}\|_F \\ &\quad + \|\Delta\mathbf{Z}\|_F \|\mathbf{Z}^{-1}\|_F + \|\Delta\mathbf{W}\|_F \end{aligned} \quad (38)$$

From equations (21) and (22), we know  $\|\mathbf{Z}^{-1}\|_F = \|\Lambda_{1/2}^{-1}\|_F$  and  $\|\mathbf{Z}'^{-1}\|_F = \|\Lambda'_{1/2}\|_F$ . Obviously, because  $\mathbf{M} = \mathbf{Z}\mathbf{Z}^T$  and  $\mathbf{M}' = \mathbf{Z}'\mathbf{Z}'^T$ , the estimation will be also very error-sensitive when the dispersion matrix  $\mathbf{M}$  or  $\mathbf{M}'$  has a close-to-zero eigenvalue. There is also an additional rotation-error term that can affect our final results.

Therefore, from equations (34) and (38), it is easy to realize that there exist some special cases whose motion parameters are inherently hard to estimate. However, readers should realize that the motion estimation of these special cases fails because the object of interest is allowed to move in 3D space in a linearly-deformable manner. For example, a moving rigid planar patch in 3D space will have an undefined expansion along its normal direction, which causes singularity.

## SIMULATION EXPERIMENTS

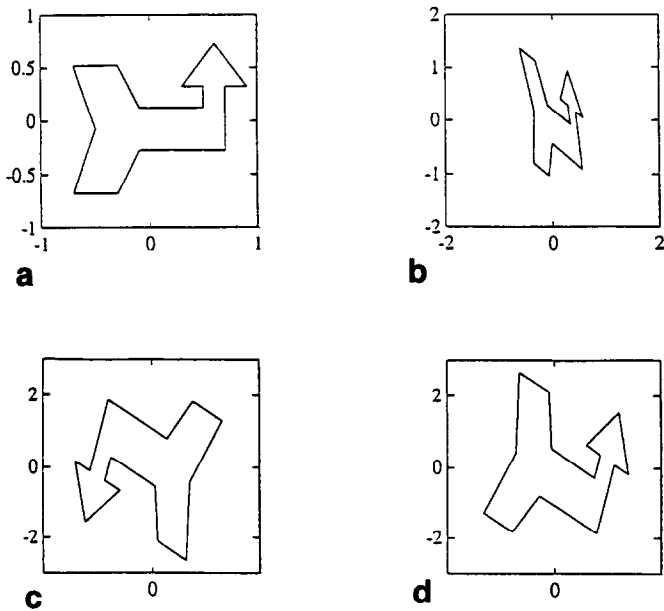
To test our derivations described above, several simulation experiments are designed here. The main objectives of these experiments are to prove that our methods are absolutely accurate when no position errors exist, to show that the error performance of the modified method is superior to the weighting method, and to show how the modified method solves the ambiguity problem.

Only position errors are considered in the following experiments.

### Error-free cases

In this experiment, two kinds of structures are tested: a 2D continuous shape and a 3D discrete-point structure. Our main objective is to show the validity of the algorithms derived above when no position errors are introduced.

Figures 9a–d show the normalization process of 2D shapes  $S$  and  $S'$ . The shape  $S$  is contained in a rectangle of  $1.6 \times 1.4$ . Figure 9a is the shape before motion ( $S$ ). Figure 9b is the shape after motion ( $S'$ ). Figure 9c is the normalized shape of  $S$ , that is  $\hat{S}$ . Figure 9d is the normalized shape of  $S'$ , that is  $\hat{S}'$ . Table 1 lists the real and estimated motion parameters. It can be easily observed that the estimation of  $\mathbf{H}$  is accurate. Notice



**Figure 9** Normalization process of a 2D continuous shape. (a) Unmoved shape  $S$ ; (b) moved shape  $S'$ ; (c) normalized shape  $\hat{S}$  of  $S$ ; (d) normalized shape  $\hat{S}'$  of  $S'$

that the name 'Weight: ( $k$ )' on this table represents that the weighting function  $g(r)$  used in the weighting method is defined as  $g(r) = r^k$ .

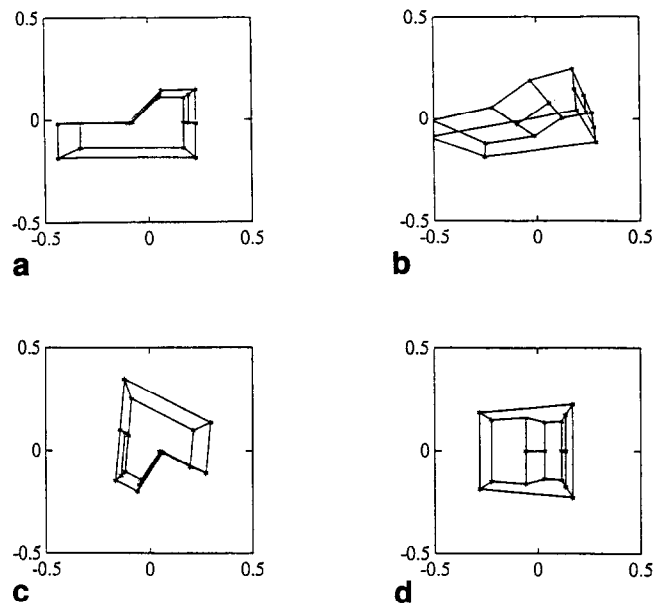
Figure 10a-d show the normalization process of a discrete-point 3D structure (only vertices are considered). The whole structure of this car-like structure is contained in a rectangle volume whose size is  $4 \times 2 \times 2$ . Figure 10a-d separately represents the object  $S$ ,  $S'$ ,  $\hat{S}$  and  $\hat{S}'$ . Table 2 lists both the real and estimated motion parameters. It can easily be observed that the estimation is accurate. Notice the name 'Weight: ( $a, b$ )' represents that the two weighting functions  $g_1(r)$  and  $g_2(r)$  used in this weighting method are separately defined as  $g_1(r) = r^a$  and  $g_2(r) = r^b$ .

**Cases when position errors exist**

In this experiment, we try to show how the modified method is superior to the weighting method in error performance. Here, we use the 3D discrete-point structure in Figure 10. Position errors are added to each point, and these errors are assumed to be uniformly distributed in a small cubic whose edge

**Table 1** Listing table of the estimated motion for Figure 9

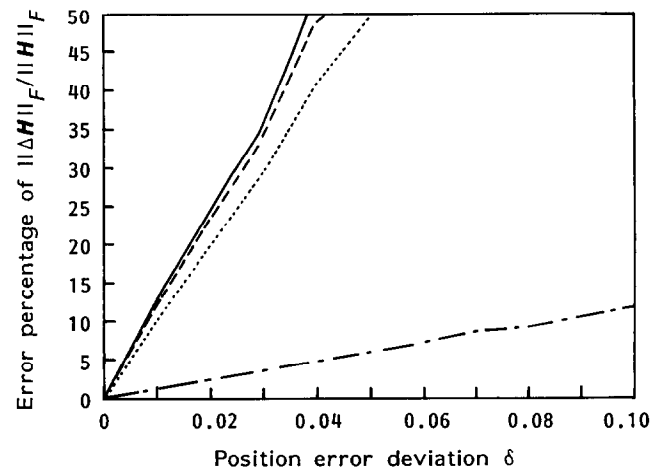
Methods	$H$ or $\hat{H}$	$\Delta H (10^{-15})$	% error
Real answer	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$	$\begin{bmatrix} 0.00 & 0.00 \\ 0.00 & 0.00 \end{bmatrix}$	0
Modified	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$	$\begin{bmatrix} 0.11 & -0.08 \\ 0.22 & -0.22 \end{bmatrix}$	$1.7 \times 10^{-14}$
Weight: (1)	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$	$\begin{bmatrix} -0.22 & 1.06 \\ -1.67 & -0.67 \end{bmatrix}$	$1.0 \times 10^{-13}$
Weight: (2)	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$	$\begin{bmatrix} -0.22 & 0.61 \\ -0.89 & -1.11 \end{bmatrix}$	$7.7 \times 10^{-14}$



**Figure 10** Normalization process of a 3D discrete-point structure. (a) Unmoved object  $S$ ; (b) moved object  $S'$ ; (c) normalized object  $\hat{S}$  of  $S$ ; (d) normalized object  $\hat{S}'$  of  $S'$

lengths are all  $2\delta$ . In Figure 11 four curves are plotted. Each point of the curves comes from the averaging of 100 different tests under the same error level  $\delta$ . The lowermost curve is the result of the modified method; the higher three curves are the results of the weighting method, whose weighting functions are  $g_1(r) = r^a$  and  $g_2(r) = r^b$ . ( $a, b$ ) represents the weighting functions in this experiment. Notice that the error percentage of the estimation is defined as  $\|\Delta H\|_F / \|H\|_F \times 100\%$ .

From the results shown in Figure 11, we find that the error performance of the modified method (assuming that correct correspondences are obtained) is considerably better than that of the weighting method. In fact, when the object structure is more asymmetric, the performance of the weighting method, may be better. This means that the structure of the object has a critical influence to the weighting method.



**Figure 11** Error performance of the weighting and modified methods. We can observe that the modified method is significantly better than the weighting method in error sensitivity



**Table 2** Listing table of the estimated motion for *Figure 10*

Methods	$H$ or $\hat{H}$	$\Delta H$ ( $10^{-14}$ )	% error
Real answer	$\begin{bmatrix} 1.1008 & -0.0101 & 0.5967 \\ 0.2069 & 0.5504 & -0.5384 \\ -0.4307 & 0.2387 & 1.2665 \end{bmatrix}$	$\begin{bmatrix} 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 \end{bmatrix}$	0
Modified	$\begin{bmatrix} 1.1008 & -0.0101 & 0.5967 \\ 0.2069 & 0.5504 & -0.5384 \\ -0.4307 & 0.2387 & 1.2665 \end{bmatrix}$	$\begin{bmatrix} 0.00 & 0.04 & 0.02 \\ -0.00 & 0.00 & 0.00 \\ 0.02 & -0.02 & -0.07 \end{bmatrix}$	$4.2 \times 10^{-14}$
Weight: (0.5, 1)	$\begin{bmatrix} 1.1008 & -0.0101 & 0.5967 \\ 0.2069 & 0.5504 & -0.5384 \\ -0.4307 & 0.2387 & 1.2665 \end{bmatrix}$	$\begin{bmatrix} -1.11 & 2.45 & 4.03 \\ 0.44 & -0.08 & 0.63 \\ -1.68 & 0.86 & -1.58 \end{bmatrix}$	$2.7 \times 10^{-12}$
Weight: (2, 1)	$\begin{bmatrix} 1.1008 & -0.0101 & 0.5967 \\ 0.2069 & 0.5504 & -0.5384 \\ -0.4307 & 0.2387 & 1.2665 \end{bmatrix}$	$\begin{bmatrix} -0.60 & 1.42 & 2.11 \\ 0.19 & 0.08 & 0.31 \\ -0.87 & 0.32 & -0.80 \end{bmatrix}$	$1.4 \times 10^{-12}$

To provide a more clear sense about the motion estimation, some numerical results are listed in *Tables 3* and *4*. Notice that the performance of the weighting method in *Table 3* is not so sensitive to position errors as that in *Table 4*, because the tested 2D shape is quite asymmetric, and only one weighting function is used in the 2D case, and the final summation vector and then the estimated rotation  $R$  will not be influenced by other summation vectors (comparing it only with the 3D case).

**Cases when ambiguities exist**

We know from earlier that the weighting method fails when ambiguity or a degenerate case happens. This

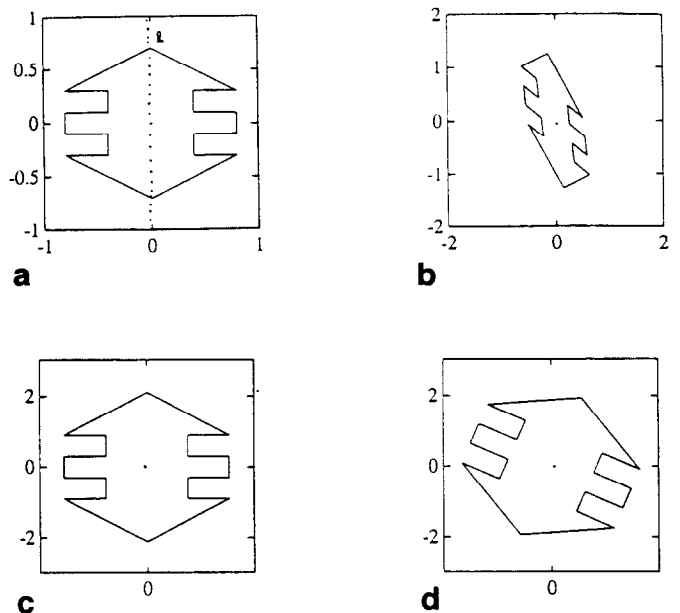
**Table 3** Estimation result of the 2D shapes displayed in *Figure 9*. Here, the error level  $\delta$  is set to 0.02

Methods	$H$ or $\hat{H}$	% error
Real answer	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$	0
Modified	$\begin{bmatrix} 0.7096 & -0.1928 \\ -0.5781 & 1.7799 \end{bmatrix}$	1.58
Weight: (1)	$\begin{bmatrix} 0.7168 & -0.1981 \\ -0.5806 & 1.7623 \end{bmatrix}$	2.24
Weight: (2)	$\begin{bmatrix} 0.7175 & -0.2081 \\ -0.5788 & 1.7651 \end{bmatrix}$	2.49

**Table 4** Estimation results of the 3D object displayed in *Figure 10*. Here, the error level  $\delta$  is set to 0.01

Methods	$H$ or $\hat{H}$	% error
Real answer	$\begin{bmatrix} 1.1008 & -0.0101 & 0.5967 \\ 0.2069 & 0.5504 & -0.5384 \\ -0.4307 & 0.2387 & 1.2665 \end{bmatrix}$	0
Modified	$\begin{bmatrix} 1.0983 & -0.0103 & 0.5964 \\ 0.2084 & 0.5500 & -0.5370 \\ -0.4332 & 0.2430 & 1.2677 \end{bmatrix}$	0.3016
Weight: (0.5, 1)	$\begin{bmatrix} 1.0869 & 0.0256 & 0.6250 \\ 0.2092 & 0.5539 & -0.5320 \\ -0.4449 & 0.2432 & 1.2561 \end{bmatrix}$	2.56
Weight: (2, 1)	$\begin{bmatrix} 1.0861 & 0.0342 & 0.6210 \\ 0.2071 & 0.5572 & -0.5327 \\ -0.4430 & 0.2381 & 1.2577 \end{bmatrix}$	2.75

experiment shows how the modified method solves the ambiguity problem. *Figure 12a-d* are similarly defined as in *Figure 9a-d*. Notice that the 2D shape we use here is symmetric to its centre. When the weighting method is used ( $g(r) = r^k$ ), both of the weighted summation vectors  $v_g$  and  $v'_g$  are zeros, which makes the weighting method fail. However, the modified method can solve two possible solutions listed in *Table 5*. One of them



**Figure 12** Normalization process of a 2D symmetric shape. (a) Unmoved shape  $S$ ; (b) moved shape  $S'$ ; (c) normalized shape  $\hat{S}$  of  $S$ ; (d) normalized shape  $\hat{S}'$  of  $S'$

**Table 5** Listing table at the estimated motion. Note the failure of the weighting method

Methods	$H$ or $\hat{H}$
Real answer	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$
Modified # 1	$\begin{bmatrix} 0.7 & -0.2 \\ -0.6 & 1.8 \end{bmatrix}$
Modified # 2	$\begin{bmatrix} -0.7 & 0.2 \\ 0.6 & -1.8 \end{bmatrix}$
Weighting	fail

corresponds to the true answer; the other is an ambiguous solution.

## CONCLUSION

In this paper, we have attempted to find the motion parameters of a uniformly-deformable object in 2D or 3D space. The features representing the object of interest are allowed to be discrete points, continuous 2D shapes or a 3D solid volume. The so-called *affine* motion model is defined as a linearly-deformable one which must be satisfied everywhere on the object (as described in equations (1) and (18)).

As we know, we often need to solve the correspondence problem before we want to estimate the motion parameters of an object. However, this problem has been hard enough in the case of rigid motion, and is even harder in the case of deformable motion. To prevent such a difficulty, we have theoretically developed a method without using one-to-one point correspondences, and a closed-form solution has been derived. The concept of normalization is the key idea in our method.

However, it is not enough to just know that this method is accurate when no position errors exist. We should notice many other important things. For example, if position errors exist, how robust are our methods? How do the structure of the object and its associated motion parameters affect the error performance? Can our method solve all kinds of object structure and motion parameters? If not, in what cases will our methods break down? If they break down, is there any other method which can solve it? All those questions are discussed and answered in our paper.

Here, we summarize our final conclusions:

1. According to the derivations discussed earlier, we know that the position error sensitivity of motion parameters depends upon the dispersion matrices of the object before and after motion. The error performance is inherently bad when there exists a close-to-zero eigenvalue of the dispersion matrices. This means that the object structure is almost degenerate, and no method can estimate its motion parameters. For example, a 2D plane object in 3D space and a 1D line object in 2D space are degenerate structures. (Notice the deformation motion model we used here.)
2. We presented the weighting method and the modified method; the former uses distance weighting of the normalized object as its main idea, while the latter uses the characters of normalized object to solve the correspondence problem. Both the theory and experimentation indicate that the error performance of the modified method is better than that of the weighting method. However, the modified method needs more time and computational cost to determine the best matching of the normalized objects.

3. The modified method (especially to the discrete-point structure) can perform as well as those methods which previously know the correct correspondences.
4. If the normalized structure of the object of interest is symmetric to its centre, the weighting method will fail because the weighted summation vector  $v_g$  (or  $V_g$ ) is always zero no matter what kind of weighting function  $g$  is used. But the modified method can solve this singular case, and provide all possible solutions.
5. Some methods using high-order moments or tensors to solve the motion parameters are in fact similar to our weighting method. We can thus predict that the error performance of such tensor methods should be comparable with the weighting method.

Although only theoretical research is presented in our paper, we can apply this result to many real applications that need point correspondences and where the objects satisfy the affine transformation. For example, if a moving rigid planar patch is placed at a distance far from the camera, its two image shapes projected on the image plane at two different time instants will approximately satisfy the 2D affine relationship, and we can estimate the correspondence by obtaining their best-fitted affine transformation. Another example is to estimate the 3D motion of the heart by obtaining the 3D data from CAT (computer-aided tomography), as done by Faber and Stokely<sup>15</sup>.

## REFERENCES

- 1 Hu, M K 'Visual pattern recognition by moment invariants', *IRE Trans. Infor. Theory*, Vol 8 (February 1962) pp 179-187
- 2 Lo, C-H and Don, H-S '3-D moment forms: their construction and application to object identification and positioning', *IEEE Trans. PAMI*, Vol 11 No 10 (October 1989)
- 3 Yaser, S, Abu-Mostafa and Psaltis, D 'Recognitive aspects of moment invariants', *IEEE Trans. PAMI*, Vol 6 No 6 (November 1984)
- 4 Cyganski, D and Orr, J A 'Applications of tensor theory to object recognition and orientation determination', *IEEE Trans. PAMI*, Vol 7 No 6 (November 1985)
- 5 Sluzek, A 'Using moment invariants to recognize and locate partially occluded 2-D object', *Patt. Recogn. Lett.*, Vol 7 (1988) pp 253-257
- 6 Sadjadi, -A and Hall, E L 'Three-dimensional moment invariants', *IEEE Trans. PAMI*, Vol 2 No 2 (March 1980)
- 7 Arbter, K, Snyder, W E, Burkhardt, H and Hirzinger, G 'Applications of affine-invariant Fourier descriptors to recognition of 3-D objects', *IEEE Trans. PAMI*, Vol 12 No 7 (July 1990)
- 8 Young, T Y and Wang, Y-L 'Analysis of three-dimensional rotation and linear shape changes', *Patt. Recogn. Lett.*, Vol 2 (1984) pp 239-242
- 9 Pinjo, Z, Cyganski, D and Orr, J A 'Determination of 3-D object orientation from projections', *Patt. Recogn. Lett.*, Vol 3 (1985) pp 351-356
- 10 Sluzek, A 'Identification of planar objects in 3-D space from perspective projection', *Patt. Recogn. Lett.*, Vol 7 (1988) pp 59-63
- 11 Blake, A and Cipolla, R 'Robust estimation of surface curvature from deformation of apparent contours', *Image & Vision Comput.*, Vol 9 No 2 (April 1991)
- 12 Leu, G 'Shape normalization through compacting', *Patt. Recogn. Lett.*, Vol 10 (1989) pp 243-250
- 13 Lee, R, Lu, P-C and Tsai, W-H 'Moment preserving detection of

- elliptical shapes in gray-scale images', *Patt. Recogn. Lett.*, Vol 11 (1990) pp 405-414
- 14 Chaudhuri, S and Chatterjee, S 'Motion analysis of a homogeneously deformable object using subset correspondences', *Patt. Recogn.*, Vol 24 No 8 (1991) pp 739-745
- 15 Faber, T L and Stokely, E M 'Orientation of 3-D structures in medical image', *IEEE Trans. PAMI*, Vol 10 No 5 (September 1988)
- 16 Pei, S C and Liou, L-G 'Tracking a planar patch in 3-D space by affine transformation in monocular and binocular vision', *Patt. Recogn.*, Vol 26 No 1 (1993) pp 23-31
- 17 Pei, S C and Liou, L-G 'Finding the motion, position and orientation of a planar patch in 3-D space from scaled-orthographic projection', *Patt. Recogn.*, Vol 27 No 1 (1994) pp 9-25
- 18 Tang, G Y 'A discrete version of Green's theorem', *IEEE Trans. PAMI* Vol 4 No 3 (May 1982)
- 19 Leu, J-G 'Computing a shape's moments from its boundary', *Patt. Recogn.*, Vol 24 No 10 (1991) pp 949-957