A New Generalized Computational Framework for Finding Object Orientation Using Perspective Trihedral Angle Constraint

Yuyan Wu, S. Sitharama Iyengar, *Senior Member, IEEE*, Ramesh Jain, *Fellow, IEEE* and Santanu Bose

***Abstract*—This paper investigates a fundamental problem of determining the position and orientation of a three-dimensional (3-D) object using single perspective image view. The technique is focused on the interpretation of trihedral angle constraint information. A new closed from solution based on Kanatani's formulation is proposed. The main distinguishing feature of our method over the original Kanatani's formulation is that our approach gives an effective closed form solution for general trihedral angle constraint. The method also provides a general analytic technique for dealing with a class of problem of shape from inverse perspective projection by using "Angle to Angle Correspondence Information." A detailed implementation of our technique is presented. Different trihedral angle configurations were generated using synthetic data for testing our approach of finding object orientation by angle to angle constraint. We performed simulation experiments by adding some noise to the synthetic data for evaluating the effectiveness of our method in real situation. It has been found that our method worked effectively in a noisy environment which confirms that the method is robust in practical application.**

***Index Terms*—Shape from angle, shape from perspective projection, pose estimation, extrinsic camera calibration, 3-D object recognition.**

I. INTRODUCTION

ONE of the major tasks in 3-D machine vision is to determine the position and orientation of a 3-D object in the scene with respect to the sensing device. For this purpose, the technology of shape from inverse perspective projection is an essential approach for model-based 3-D reconstruction. Continuing advances in the problem have derived many efficient results for the approach. There are also many applications of this approach in Robotics, Cartography and Photogrammetry, as well as in computer vision. A broader presentation on these application aspects can be found in the reference papers [8]–[13].

A. Statement of the Problem

The formal definition for the general problem of shape from inverse perspective projection can be stated as follows: Let perspective projection be the ideal model of a camera, then, the fundamental imaging process of a camera is given by

$$\begin{bmatrix} \vec{p} \\ 1 \end{bmatrix} \equiv \begin{bmatrix} k_u f & 0 & u_0 & 0 \\ 0 & k_u f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & \vec{T} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \vec{P} \\ 1 \end{bmatrix} \tag{1}$$

where, $\vec{P} = (x, y, z)^T$ is the description of a 3-D point in an object coordinate system and $\vec{p} = (u, v)^T$ is the 2-D projection of $\vec{P}$ on the image plane; where rotation R and translation $\vec{T}$ form the transformation from the object coordinate system to the camera coordinate system; f, k_u, k_v, u_0, v_0 are the intrinsic parameters of the camera. Now, suppose that certain 3-D geometric features of an object are given in an object coordinate system and their corresponding 2-D image geometric features are located in an image plane by a single perspective view. The problem of shape from inverse perspective projection is to determine the unknown rotation matrix R and the translation vector $\vec{T}$. Equivalently, the problem can also be restated that to find the pose or orientation of these 3-D geometric features in the camera coordinate system. Therefore, this approach is also named as *pose estimation* or *extrinsic camera calibration* in literature.

More specifically, three types of situations are mostly discussed in the problem of shape from inverse perspective projection.

1) Perspective point to point correspondence problem. This problem is usually called *Perspective n-point problem* or **PnP** problem [8] when n pairs of corresponding points are known.
2) Perspective line to line correspondence problem. Like the case in the above, we call the problem as **PnL** problem when n pairs of corresponding lines are specified.
3) Perspective angle to angle correspondence problem. We name this problem as **PnA** problem if n pairs of corresponding angles are given.

To achieve simplicity, stability and speed for solving an inverse perspective projection problem, a closed form solution is the most desirable result for each of PnP, PnL, and PnA problems. In this paper, a closed form solution is presented for the general problem of trihedral angle constraint, which is an P3A problem. It is a common and typical case among the PnA problems. Fig. 1 shows different viewing effects about a trihedral angle when we observe a real scene. Our method is capable of dealing with these different viewing effects.

Manuscript received July 7, 1992; revised December 8, 1993. This work was supportedd in part by ONR Grant N00014-91-J-1306. Recommended for acceptance by Associate Editor R. Nevatia.

Y. Wu, S. Iyengar, and S. Bose are with the Robotics Research Laboratory, Department of Computer Science, Louisiana State University, Baton Rouge, LA 70803 USA.

R. Jain is with the Department of Electrical and Computer Engineering, University of California-San Diego, La Jolla, CA 92093 USA.

IEEE Log Number 9405243.

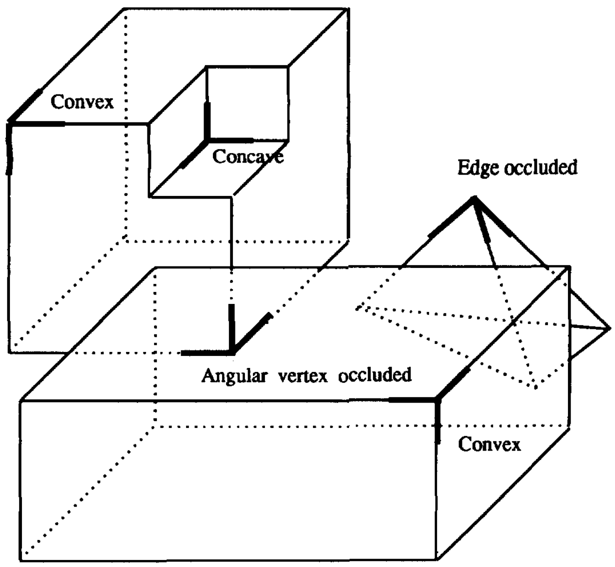


Fig. 1. Trihedral angle configurations.

B. Review of Literature

The problem of finding closed form solutions for inverse perspective projection is found in literature, and analytical solutions have been provided for 3 point correspondence (**P3P**), ([8], [9]), 4 point correspondence (**P4P**) ([8, [11]) and 3 line correspondence (**P3L**) ([12], [13]). In Section III, we can see that a linear solution may be available for point to point correspondence (**PnP**) when n is greater than or equal to 6, or for line to line correspondence (**PnL**) when n is greater than or equal to 8. However, up till now, we have not found analytical solutions for any angle to angle correspondence or (**PnA**) problem.

Among the (**PnA**) problems, trihedral angle constraint is the basic and most encountered case in practice. In recent years, trihedral angle constraint has been addressed by many authors from different viewpoints. The relevant presentations can be divided into following two categories:

a) Direct Approach: In this category, angle information is usually employed directly. Kanade [6] proposes an analytic solution for the problem under orthographic projection. For perspective projection, algebraic solutions have been given for special cases when two or three space angles are right angles by Kanatani [3], [15], Shakunaga and Kaneko [5]; in addition, some constructive algorithms are suggested for solving the general problem by Horaud [7], Shakunaga and Kaneko [5]; but further results in this category are not found in literature.

b) Indirect Approach: Geometrically, without employing angles directly, the configuration of a trihedral angle can also be specified by four space points or by a junction of three 3-D lines. In this sense, we can consider trihedral angle constraint as a special case of the P4P problem or the P3L problem. Therefore, the methods for solving these two types of problems can be applied for trihedral angle constraint ([11], [12]). Because the angle information is not used explicitly by the methods in this category, we call it indirect approach.

C. A New Direct Solution for Trihedral Angle Constraint

Our new solution for trihedral angle constraint uses the direct approach. Based on the original presentation scheme for the problem proposed by Kanatani [3], a complete analytic solution is developed. Compared with previous works in this direction, the main distinguishing feature of our method is it makes the trihedral angle constraint can be easily used for general situation. The method can also be considered as a closed form solution for the general PnA problems in a minimal condition. Here the angle information is effectively and directly used for the problem of shape from inverse perspective projection.

There are significant differences distinguishing our approach from the methods of P4P [11] and P3L [12] which use the Indirect approach. In brief, notice that the angle measure is independent of the coordinate system; but the description of a point or a line is dependent on a coordinate system and so it varies when the related coordinate system is changed. This is the distinguishing feature of the angle constraint compared to the point constraint or the line constraint. Therefore, our method possesses its special advantage and usages in different application situations.

In Section II, our method will be developed in detail. Then, in Sections III and IV, broad discussions and the results of simulation experiments will be presented.

II. A New Mathematical Framework

A. Preliminary Formulation

The coordinate systems considered in the paper are right handed orthogonal systems. According to the common model of the perspective projection, the following three coordinate systems are related to our problem (Fig. 2).

1) The object coordinate system is a local 3-D coordinate system used for defining objects.
2) The camera coordinate system is the 3-D coordinate system attached to a camera. We assume that the origin of the coordinate system is the center of projection, and its z-axis is the view axis.
3) The image coordinate system is the projection plane. It is specified within the camera coordinate system by centering at the point $(0, 0, f)$ and its two axes, u-axis and v-axis are parallel to the x-axis and y-axis of the camera coordinate system, respectively. f is the focal length of the camera.

Canonical Image Structure: We can rewrite the expression (1) of imaging transformation as

$$\begin{bmatrix}\vec{p}\\1\end{bmatrix} \equiv (KR\ K\vec{T})\begin{bmatrix}\vec{p}\\1\end{bmatrix} \quad with\ K = \begin{bmatrix}k_u f & 0 & u_0\\ 0 & k_v f & v_0\\ 0 & 0 & 1\end{bmatrix}$$

For the problem of shape from inverse perspective projection, we assume that the intrinsic parameters of a camera model are given. Therefore, we can derive

$$\begin{bmatrix}\vec{p}'\\1\end{bmatrix} \equiv K^{-1}\begin{bmatrix}\vec{p}\\1\end{bmatrix} \equiv K^{-1}(KR\ K\vec{T})\begin{bmatrix}\vec{p}\\1\end{bmatrix} = (R\vec{T})\begin{bmatrix}\vec{p}\\1\end{bmatrix} \quad (2)$$

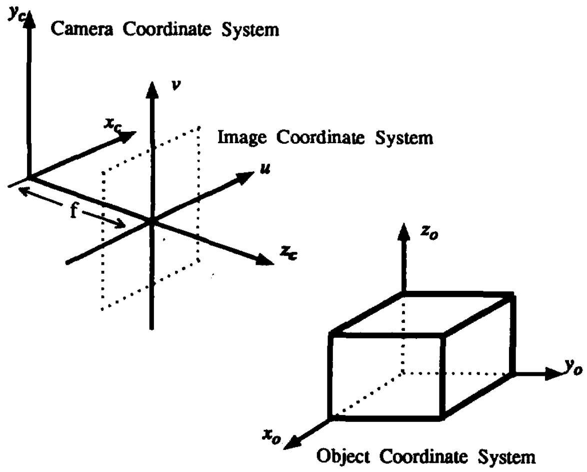


Fig. 2. An illustration of three types of coordinate systems.

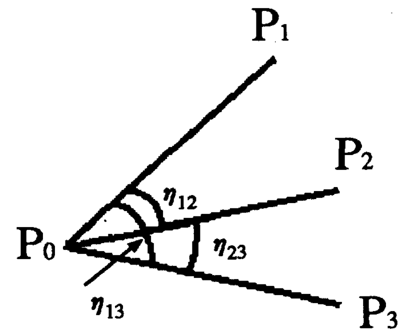


Fig. 3. Definition of a trihedral angle.

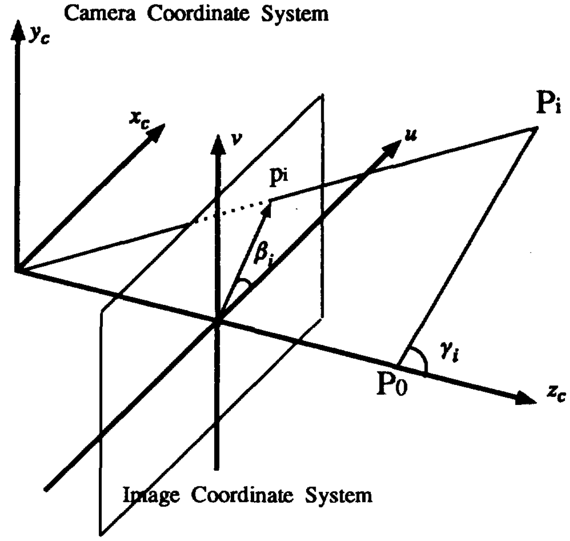


Fig. 4. Angle's definition for trihedral angle constraints.

where $\vec{p'}$ is determined only by the extrinsic parameters of rotation R and translation $\vec{T}$; we may regard it as a non-digitalized perspective projection of $\vec{P}$ with focal length $f = 1$ and call it as Canonical Image. For our problem of finding shape from trihedral angle constraint, since canonical image is much more convenient than the original digitalized image and it is always available, we will mainly consider the canonical representation in the following discussions.

View Orientation Transformation Schemes: We define the view orientation transformation as a pure rotation transformation upon a camera coordinate system. Suppose the rotation $R = (r_{ij})_{3x3}$ defines a view orientation transformation such that $\vec{P'} = R\vec{P}$. Then, the corresponding relationship between the two image points of $\vec{p}$ and $\vec{p'}$ is uniquely determined under the transformation by

$$\begin{aligned} u' &= \frac{x'}{z'} = \frac{r_{11}u + r_{12}v + r_{13}}{r_{31}u + r_{32}v + r_{33}} \\ u &= \frac{x}{z} = \frac{r_{11}u' + r_{21}v' + r_{31}}{r_{13}u' + r_{23}v' + r_{33}} \\ v' &= \frac{y'}{z'} = \frac{r_{21}u + r_{22}v + r_{23}}{r_{31}u + r_{32}v + r_{33}} \\ v &= \frac{y}{z} = \frac{r_{12}u' + r_{22}v' + r_{32}}{r_{13}u' + r_{23}v' + r_{33}}. \end{aligned} \tag{3}$$

This relation will be used to facilitate problem formulation of trihedral constraint. Notice that for an arbitrary view orientation, there are infinite view orientation transformations which can transform the view axis of a camera coordinate system from an old orientation to a new one. In this paper, we just consider the view orientation transformation which is formed by a rotation around the y-axis of camera then followed by a rotation around the x-axis of camera. This choice comes from the simulation for the normal situations when people turn their viewing orientation from one point to another. Let a new view orientation be selected by a image point $(u, v)^T$, then, the rotation matrix which turns the z-axis of a camera from its old orientation to the new one can be determined as below:

$$R = \begin{bmatrix} 1/d_1 & 0 & -u/d_1 \\ -uv/d_1 d_2 & d_1/d_2 & -v/d_1 d_2 \\ u/d_2 & v/d_2 & 1/d_2 \end{bmatrix} \tag{4}$$

where, $d_1 = \sqrt{u^2 + 1}$, $d_2 = \sqrt{u^2 + v^2 + 1}$. The matrix R turns the z-axis from the old orientation to the new orientation.

Trihedral Angle Constraint Formulation: Consider a trihedral angle in Fig. 3. The trihedral angle is formed by $\vec{P}_i = (x_i, y_i, z_i)^T, i = 0, 1, 2, 3$, with $\vec{P}_0$ being its angular point. Denote $\vec{p}_i = (u_i, v_i)^T$ to be the perspective projection of $\vec{P}_i$. Because a view orientation transformation can always be employed to turn the view axis of camera to pass through $\vec{p}_0$ and so $\vec{P}_0$, then, without loss of generality, we assume that $\vec{P}_0$ is located on the view axis. As shown in Fig. 4, let β_i be the angle formed by $\vec{p}_i$ and the u-axis, $-\pi < \beta_i < \pi; \vec{L}_i = \vec{P}_i - \vec{P}_0, \vec{N}_i$ be the unit direction vector of $\vec{L}_i; \gamma_i$ be the angle formed by L_i and the view axis, $0 < \gamma_i < \pi$; then, we have

$$\sin\beta_i = \frac{v_i}{\sqrt{u_i^2+v_i^2}} \quad \cos\beta_i = \frac{u_i}{\sqrt{u_i^2+v_i^2}}$$
$$\vec{N}_i = (\sin\gamma_i\cos\beta_i, \sin\gamma_i\ \beta_i, \cos\gamma_i)^T. \tag{5}$$

Now, suppose η_{ij} represents the angle formed by $\vec{N}_i$ and $\vec{N}_j$, we have the angle constraint

$$\vec{N}_i \bullet \vec{N}_j = \sin\gamma_i\sin\gamma_j\cos(\beta_i-\beta_j) + \cos\gamma_i\cos\gamma_j = \cos\eta_{ij}. \tag{6}$$

It follows that the constraint for trihedral angle can be written as

$$\begin{aligned}\vec{N}_1 \bullet \vec{N}_2 &= \sin\gamma_1\sin\gamma_2\cos(\beta_1-\beta_2) + \cos\gamma_1\cos\gamma_2 = \cos\eta_{12}\\ \vec{N}_1 \bullet \vec{N}_3 &= \sin\gamma_1\sin\gamma_3\cos(\beta_1-\beta_3) + \cos\gamma_1\cos\gamma_3 = \cos\eta_{13}\\ \vec{N}_2 \bullet \vec{N}_3 &= \sin\gamma_2\sin\gamma_3\cos(\beta_2-\beta_3) + \cos\gamma_2\cos\gamma_3 = \cos\eta_{23}.\end{aligned} \tag{7}$$

When the three angles η_{12}, and η_{13}, and η_{23} are given, we have a system of three equations and three unknowns. So we expect to solve $\gamma_1, \gamma_2, \gamma_3$ and then to determine the orientation of the trihedral angle in camera coordinate system.

Kanatani [3] first suggests the formulation for angle constraint. The advantage of this formulation is that the expressions are simple by moving the vertex of a trihedral angle on the view axis. The solution of (7) has been addressed by Kanatani [3], [15] for the special case where at least two of η_{12}, η_{13}, and η_{23} are right angles.

We will now derive a complete solution for (7).

B. An Analytical Solution for the Trihedral Angle Constraint

Estimate the Orientation: Our idea for solving (7) is straightforward. First, assume that $\vec{N}_3$ can be expressed by $\vec{N}_1$ and $\vec{N}_2$ as

$$\vec{N}_3 = a\vec{N}_1 + b\vec{N}_2 + c\vec{N}_1 \times \vec{N}_2. \tag{8}$$

We have

$$\begin{aligned}\vec{N}_1 \bullet \vec{N}_3 &= a + b\cos\eta_{12} = \cos\eta_{13}\\ \vec{N}_2 \bullet \vec{N}_3 &= a\cos\eta_{12} + b = \cos\eta_{23}\\ \vec{N}_3 \bullet \vec{N}_3 &= a\cos\eta_{13} + b\cos\eta_{23} + c^2\sin^2\eta_{12} = 1.\end{aligned}$$

Then, the coefficients a, b and c can be derived

$$\begin{aligned}a &= (\cos\eta_{13} - \cos\eta_{12}\cos\eta_{23})/\sin^2\eta_{12}\\ b &= (\cos\eta_{23} - \cos\eta_{12}\cos\eta_{13})/\sin^2\eta_{12}\\ c &= \pm\sqrt{(1 - a\cos\eta_{13} - b\cos\eta_{23})/\sin^2\eta_{12}}.\end{aligned}$$

Using the values of a, b and c, we can rewrite (8) to obtain (9) found at the bottom of the page. Without loss of generality, suppose $(\beta_1 - \beta_2) \neq 0$. Then,

$$\sin\gamma_1\sin\gamma_2 = \frac{\cos\eta_{12} - \cos\gamma_1\cos\gamma_2}{\cos(\beta_1-\beta_2)} \tag{10}$$

by the first equation of (7). Substituting (9) and (10) into the second and third equations of (7), we get

$$\begin{aligned}&A_1\cos^2\gamma_1 + B_1\cos\gamma_1\cos\gamma_2 + C_1\cos^2\gamma_1\cos\gamma_2\\ &\quad + D_1\cos\gamma_1 + E_1\cos\gamma_2 + F_1 = 0\\ &A_2\cos^2\gamma_2 + B_2\cos\gamma_2\cos\gamma_1 + C_2\cos^2\gamma_2\cos\gamma_1\\ &\quad + D_2\cos\gamma_2 + E_2\cos\gamma_1 + F_2 = 0.\end{aligned} \tag{11}$$

Note that both equalities in (9) for $\sin\gamma_3$ yield the same result described by (11), where

$$\begin{aligned}A_1 &= a\cos(\beta_1-\beta_2)\sin(\beta_1-\beta_2)\\ B_1 &= b\sin(\beta_2-\beta_3)\\ C_1 &= c\cos(\beta_2-\beta_1)\sin(\beta_3-\beta_1)\\ D_1 &= -c\cos(\beta_2-\beta_3)\cos\eta_{12}\\ E_1 &= c\cos(\beta_1-\beta_2)\cos(\beta_1-\beta_3)\\ F_1 &= a\cos(\beta_1-\beta_3)\sin(\beta_2-\beta_1)\\ &\quad + \sin(\beta_3-\beta_2)\cos\eta_{13}\end{aligned} \tag{12}$$

and

$$\begin{aligned}A_2 &= b\cos(\beta_1-\beta_2)\sin(\beta_2-\beta_3)\\ B_2 &= a\sin(\beta_1-\beta_3)\\ C_2 &= c\cos(\beta_2-\beta_1)\sin(\beta_3-\beta_2)\\ D_2 &= c\cos(\beta_1-\beta_3)\cos\eta_{12}\\ E_2 &= -c\cos(\beta_1-\beta_2)\cos(\beta_2-\beta_3)\\ F_2 &= b\cos(\beta_2-\beta_3)\sin(\beta_1-\beta_2)\\ &\quad + \sin(\beta_3-\beta_1)\cos\eta_{23}.\end{aligned} \tag{13}$$

From the first equation of (11), we have

$$\cos\gamma_2 = -\frac{A_1\cos^2\gamma_1 + D_1\cos\gamma_1 + F_1}{C_1\cos^2\gamma_1 + B_1\cos\gamma_1 + E_1}. \tag{14}$$

Replacing $\cos\gamma_2$ by (14) in the second equation of (11), it follows that

$$\begin{aligned}&s_5\cos^5\gamma_1 + s_4\cos^4\gamma_1 + s_3\cos^3\gamma_1\\ &\quad + s_2\cos^2\gamma_1 + s_1\cos\gamma_1 + s_0 = 0\end{aligned} \tag{15}$$

$$\begin{aligned}\sin\gamma_3 &= \frac{a\sin\gamma_1\cos\beta_1 + b\sin\gamma_2\cos\beta_2 + c(\sin\gamma_1\sin\beta_1\cos\gamma_2 - \sin\gamma_2\sin\beta_2\cos\gamma_1)}{\cos\beta_3}\\ \sin\gamma_3 &= \frac{a\sin\gamma_1\sin\beta_1 + b\sin\gamma_2\sin\beta_2 + c(\sin\gamma_2\cos\beta_2\cos\gamma_1 - \sin\gamma_1\cos\beta_1\cos\gamma_2)}{\sin\beta_3}\\ \cos\gamma_3 &= a\cos\gamma_1 + b\cos\gamma_2 + c\sin\gamma_1\sin\gamma_2\sin(\beta_2-\beta_1).\end{aligned} \tag{9}$$

where

$$
\begin{aligned}
s_5 &= C_2A_1^2 + E_2C_1^2 - B_2A_1C_1 \\
s_4 &= A_2A_1^2 + F_2C_1^2 - B_2A_1B_1 - B_2D_1C_1 \\
&\quad - D_2A_1C_1 + 2C_2A_1D_1 + 2E_2C_1B_1 \\
s_3 &= 2A_2A_1D_1 - D_2A_1B_1 - D_2C_1D_1 + 2E_2C_1E_1 \\
&\quad + 2C_2A_1F_1 + C_2D_1^2 - B_2A_1E_1 - B_2CF_1 \\
&\quad - B_2D_1B_1 + 2F_2C_1B_1 + E_2B_1^2 \\
s_2 &= 2A_2A_1F_1 + A_2D_1^2 - D_2A_1E_1 - D_2F_1C_1 \\
&\quad - D_2D_1B_1 + 2F_2C_1E_1 + F_2B_1^2 + 2C_2D_1F_1 \\
&\quad - B_2D_1E_1 - B_2B_1F_1 + 2E_2B_1E_1 \qquad (16) \\
s_1 &= 2A_2D_1F_1 - D_2D_1E_1 - D_2F_1B_1 + 2F_2B_1E_1 \\
&\quad + C_2F_1^2 - B_2F_1E_1 + E_2E_1^2 \\
s_0 &= A_2F_1^2 + F_2E_1^2 - D_2F_1E_1.
\end{aligned}
$$

By (15), (14) and the third equality of (9), we can solve the $\cos\gamma_1, \cos\gamma_2$ and $\cos\gamma_3$ step by step. The position of a trihedral angle is determined in camera coordinate system by the following expressions (referring to Fig. 4):

$$\vec{P}_i = \vec{P}_0 + l_i\vec{N}_i \qquad (i = 1, 2, 3) \qquad (17)$$

where $l_i > 0$ is the length of $\vec{L}_i = \vec{P}_i - \vec{P}_0$. Because so far only the three unit vectors $\vec{N}_i$ can be assigned by solving (7), we only obtain the orientation of a trihedral angle. To find its full position, more information is necessary.

Determine the Full Position of a Trihedral Angle: Suppose that a trihedral angle in an object coordinate system is given by:

$$\vec{P'}_i = \vec{P'}_0 + l_i'\vec{N}_i' \qquad (i = 1, 2, 3) \qquad (18)$$

where the corresponding relationship between (17) and (18) is specified by $\vec{P}_i$ to $\vec{P'}_i$, $\vec{N}_i$ to $\vec{N'}_i$ and $l_i = l_i'$. To calculate the coordinate transformation $\vec{P}_i = R\vec{P'}_i + \vec{T}$ from the object frame to the camera frame, let $\vec{P'}$ and $\vec{p}$ be a pair of matched object point and image point; then, denote $\vec{P'} = (x', y', z')^T, \vec{p} = (u, v)^T, R = (r_{ij})_{3\times3}, \vec{T} = (t_x, t_y, t_z)^T$, we have

$$
\begin{aligned}
&r_{11}x' + r_{12}y' + r_{13}z' + t_x \\
&\quad - u(r_{31}x' + r_{32}y' + r_{33}z' + t_z) = 0 \\
&r_{21}x' + r_{22}y' + r_{23}z' + t_y \\
&\quad - v(r_{31}x' + r_{32}y' + r_{33}z' + t_z) = 0. \qquad (19)
\end{aligned}
$$

The rotation matrix R can be easily found by the relation $\vec{N}_i = R\vec{N'}_i$ $(i = 1, 2, 3)$. Therefore, if two pairs of matched points are available, the translation $\vec{T}$ can be obtained by solving (19). It follows that to get a full solution for trihedral constraint, we still need two pairs of matched object point and image point.

Alternatively, if one of length l_i in (17) is known, the $\vec{P}_0$ can be simply determined by each of the following two equations provided the denominator is not zero.

$$
\begin{aligned}
z_0 &= l_i(\sin\gamma_i\cos\beta_i - u_i\cos\gamma_i)/u_i \\
z_0 &= l_i(\sin\gamma_i\sin\beta_i - v_i\cos\gamma_i)/v_i
\end{aligned}
\qquad (z_0 > 1). \qquad (20)
$$

Then, the trihedral angle is completely determined in camera frame but do not need to refer any object coordinate system.

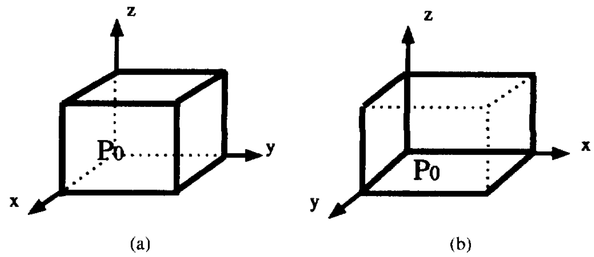


Fig. 5. Necker's cube illusion. (a) The P_0 is in the back of the cube. (b) The point P_0 is in the front of the cube.

We are more interested in (20) than (19) for our method because both the measures of angle and length are independent of a concrete coordinate system. This feature makes our method more flexible in application than the approaches of P4P [11] and L3L [12] which need to refer to some object frame.

An Algorithmic Framework for Trihedral Angle Constraint: To sum up, we list the steps of the solution procedure for the shape from trihedral angle constraint as below:

Prerequisite: Suppose that the intrinsic parameters of the camera are given; and, a trihedral configuration is picked from the image plane and the three corresponding 3-D angles have been specified.

Step 1: Use (2) to get the canonical representation for the image features.

Step 2: Use the angular vertex of the 2-D trihedral configuration to compute the rotation matrix R defined by (4); Then, transform the image coordinates from p_i's to p_i''s by using the relation in (3).

Step 3: Match the 2-D vs. 3-D angles; then, determine the original equation system by (5) and (7).

Step 4: Derive the fifth-order equation (15), then solve (15) to get $\cos\gamma_1$; if there is no solution, go to step 8.

Step 5: Calculate $\cos\gamma_2$ by (14); if there is no solution, go to step 8.

Step 6: Calculate $\cos\gamma_3$ according to the third equality of (9); if there is no solution, go to step 8;

Step 7: Check the solution against the original equation system (7).

Step 8: If there is no solution but some other matching pattern exists for the 2-D and 3-D angles, adopt a new matching pattern, go to step 3; otherwise, terminate.

Step 9: If additional information is available for finding a full solution, find the solution using (19) or (20).

Step 10: Transform the final result (17) back to the original camera coordinate system by using the inverse of the rotational matrix R defined in Step 2.

III. Analysis on the Solution of Trihedral Angle Constraint

A. The Mirror Solution

When a trihedral angle is specified just by the three angles η_{12}, η_{13}, and η_{23}, an important phenomena is the well-known "Necker's cube vision illusion." For instance, in Fig. 5, a projection of a cube wireframe may have two different explanations depending on whether we think the vertex $\vec{P}_0$ is at the back of the cube or in front of it.

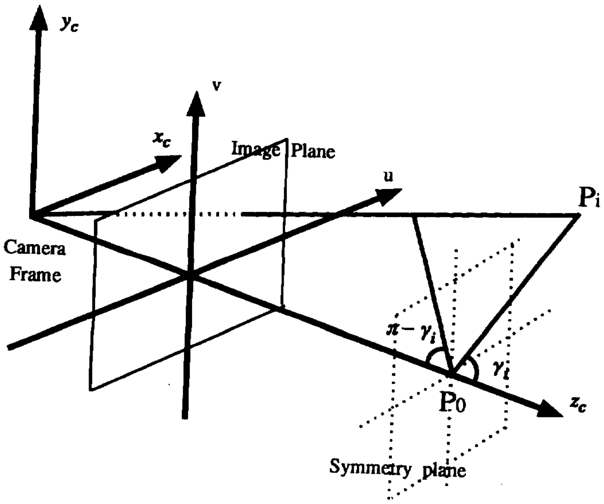


Fig. 6. Mirror solution.

The trihedral angle formulation (7) given by Kanatani [3] presents a mathematical explanation for the "Necker's cube illusion." In detail, note that if $\cos\gamma_1, \cos\gamma_2 \cos\gamma_3$, or $\gamma_1, \gamma_2, \gamma_3$ form a solution of a trihedral angle constraint, then, $-\cos\gamma_1, -\cos\gamma_2, -\cos\gamma_3$ or $\pi-\gamma_1, \pi-\gamma_2, \pi-\gamma_3$ form another solution of the same trihedral angle constraint. These two solutions are symmetric to the plane which contains $\vec{P}_0$ and are parallel to the projection plane (Fig. 6). So, we call them as mirror solutions.

In our solution scheme, notice that in (8), the coefficient c has a degree of freedom since it can take different signs. Consequently, the signs of C_1, D_1E_1, in (12) and C_2, D_2E_2 in (13) vary according to the choice of c; and so do the coefficients s_5, s_3, and s_1 in (15). Therefore, suppose that $\cos\gamma_1$ is a root of (15) when c takes a certain sing, the $-\cos\gamma_1$ must be a root of (15) when c takes another sign. Similar conclusions also hold for $\cos\gamma_2$ and $\cos\gamma_3$. In other words, the different signs of c correspond to two solution groups of (7) by mirror characteristic.

If no further clue is available, two mirror solutions are all the possible solutions for a trihedral angle constraint. However, in all cases, for a given trihedral angle constraint, the solution procedure needs to be executed just once by taking an arbitrary sign for c in (8), and then using the mirror feature to find the other solutions. The mirror feature enables us to save half the computation.

The paper [12] claims that the "Necker's cube illusion" can be suppressed in perspective projection but the conclusion is based on the prerequisite condition for the L3L approach. Our presentation shows that this phenomena exists in perspective projection as well as in orthographic projection.

B. Special Configuration Cases

Some special configurations of trihedral angle are commonly encountered in real applications. For these cases, the general quintic equation (15) can be simplified to certain lower and more succinct patterns to facilitate the solving procedures.

Coplanar Configuration: Coplanar configuration means that the three vectors $\vec{N}_1, \vec{N}_2, \vec{N}_3$ are located on a plane. In this case, we have $c = 0$ in (8). It follows that $C_i = D_i = E_i = 0 (i = 1, 2)$ in (12) and (13), and so $s_5 = s_3 = s_1 = 0$ in (15). Then, equation (15) becomes

$$s_4 \cos^4\gamma_1 + s_2 \cos^2\gamma_1 + s_0 = 0.$$

This is actually a quadratic equation on $\cos^2\gamma_1$

The Configuration With Two or Three Right Angles: Suppose there are at least two right angles in a trihedral angle. In this case, we can choose $\vec{N}_3$ such that $\vec{N}_1.\vec{N}_3 = 0$ and $\vec{N}_2.\vec{N}_3 = 0$. This implies that $a = 0$ and $b = 0$ in (8). Consequently, we have $A_i = B_i = C_i = 0 (i = 1, 2)$ in (12) and (13), and therefore $s_4 = s_2 = s_0 = 0$ in (15). Then, (15) can be rewritten as

$$s_5 \cos^4\gamma_1 + s_3 \cos^2\gamma_1 + s_1 = 0$$

As in case (a), we obtain a quadratic equation on $\cos^2\gamma_1$.

In addition to the two cases for spatial angles, certain image configurations may also decrease the order of (15). We are interested in the conditions which lead $s_5 = 0$ or $s_0 = 0$ and so a quadrinomial or a cubic can be resulted.

Special Image Configurations: Assume one right angle exists, say $\beta_1 - \beta_2 = \pi/2$, we have $A_i = E_i = 0 (i = 1, 2)$ in (12) and (13) so that $s_5 = s_0 = 0$ in (15), we get a cubic from (15) as

$$s_4 \cos^3\gamma_1 + s_3 \cos^2\gamma_1 + s_2 \cos\gamma_1 + s_1 = 0.$$

If there are three collinear points in an image configuration, let $\beta_1 - \beta_2 = \pi$, we have $C_1 = C_2 = 0$ in (12) and (13) so that $s_5 = 0$ in (15) and so (15) becomes a quadrinomial

$$s_4 \cos^4\gamma_1 + s_3 \cos^3\gamma_1 + s_2 \cos^2\gamma_1 + s_1 \cos\gamma_1 + s_0 = 0.$$

The two special image configurations may appear when some objects are overlapping each other. This is not unusual in multiple-object scene. An instance is an object which is assembled by several parts.

C. Comparison and Comments on the Problems of PnP, PnL, and PnA

The constraints for the problems of PnP, PnL, and PnA can be divided into two categories. The first category is linear constraint. In this category, for 2-D image features, the corresponding 3-D features are defined in an object coordinate system, and the transformation from the object coordinate system to the camera coordinate system is the unknown. The second category is nonlinear constraint. In this category, for the interested 2-D image features, the corresponding 3-D features are given by a group of scalars, and the unknowns define a 3-D configuration in camera coordinate system.

For example, PnL is a typical constraint in the first category because it is necessary to refer to some coordinate system for specifying a 3-D line. Concretely, let a spatial line be depicted as $\vec{q} + t\vec{m}$ in an object coordinate system, where $\vec{q}$ is a point on the line and $\vec{m}$ is the direction vector of the line; suppose the related image line is represented as $\vec{n} \cdot (u, v, 1)^T = 0$, where vector $\vec{n} = (a, b, c)$; denote the transformation from the

TABLE I
A COMPARISON OF OUR METHOD TO THE EXISTING METHODS

Type of Data Correspondence	Type of Constraint	Dependence on Object Frame	Linear Solution	Investigations on Closed Form Solution
Point to Point Correspondence	Linear	Yes	n ≥ 6	P4P, Horaud, [11], 1989
	Nonlinear	No	No	P3P & P4P, Fischler, [8], 1981 P3P, Linnainma, [9], 1988
Line To Line Correspondence	Linear	Yes	n ≥ 8	P3L, Dhome, [12], 1989 P3L, Chen, [13], 1990
Angle To Angle Correspondence	Nonlinear	No	No	P3A, This Paper 1992

object coordinate system to the camera coordinate system by rotation R and translation $\vec{T}$. Then, we get two linear constraint equations

$$\vec{n} \cdot R\vec{m} = 0 \qquad \vec{n} \cdot (R\vec{q} + \vec{T}) = 0 \tag{21}$$

for a line to line projection. PnP (19) is another constraint in the first category because the constraint equations (19) are linear, the 3-D points in (19) are defined in an object frame and the unknowns are the components of the rotation R and the translation T which form the transformation from the object coordinate system to the camera coordinate system.

Contrary to PnL, PnA is a typical constraint in the second category because only n scalars are needed for specifying n spatial angles. On the other hand, obviously (6) is nonlinear and the unknowns define the orientations of the spatial angle in camera coordinate system. An interesting fact is PnP can be presented in both categories when $n > 1$. To make the matter clear, let $\vec{P}_i = (x_i, y_i, z_i)^T (i = 1, 2)$ be two given 3-D points with the coordinates in a camera frame, we have

$$(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2 = d_{12}^2.$$

Because $x_i = u_i z_i$ and $y_i = v_i z_i$, the expression can be rewritten as

$$a_1 z_1^2 + 2b_{12} z_1 z_2 + a_2 z_2^2 = d_{12}^2 \tag{22}$$

where $a_i = u_i^2 + v_i^2 + 1, b_{12} = u_2 u_1 + v_1 v_2 + 1$. This time, we find that the PnP constraint (22) is a nonlinear equation. The 3-D features of $\vec{P}_1$ and $\vec{P}_2$ are given by their distance d_{12} and the unknowns are the z coordinate of the two points in camera coordinate system.

Generally speaking, a PnP or PnA problem can be restated as a related PmL problem, where m may differ from n. However, in mathematics, this is not the case when a PnP or PnA constraint is specified in category two. A constraint in category one may be changed into category two provided it is essentially a PnP or PnA constraint, but by no means a constraint in category two can be changed into category one. Therefore, solution approach for the constraint in category two is more powerful than an approach for the constraint in category one in dealing with a same problem.

The important facts on the problems of PnP, PnL and PnA are listed in Table I. Our approach presents the first closed form solution for P3A problem. Furthermore, by (6) we see that each pair of corresponding 2-D and 3-D angles results in a new constraint equation with two variable. That means an PnA problem is solvable in a closed from one if $n > 3$ and each spatial angle at least is a trihedral vertex. So the number of the constraint equations must be greater than or equal to the number of the unknowns. When this condition is satisfied, our approach provides a basic method to cope with the problem. Its distinctive power is that angle information is sufficient for the method.

IV. EXPERIMENTAL VALIDATION

A. Experimental Design

In regard to the application of the new developed approach, we are mainly concerned about its effects on the following three aspects.

1) Because the solutions are derived originally from the fifth order polynomial (15), there may exist at most five pairs of mirror solution. However, there are possible extraneous roots caused by the elimination process. So each solution coming from (15), (14) and (7) should be formally checked by the three inherent criteria.

   C-1: Each solution obtained by (15), (14) and (9) must be in $[-1, 1]$.

   C-2: Each group of solutions should satisfy the original equation system (7).

   C-3: If additional information about the 3-D length of the leg of a trihedral angle is available, the solution of (20) should be bigger that 0.

   In this section, our first task is to investigate how many solution can occur for an arbitrary trihedral angle constraint and whether the true solution is always obtainable by our method.

2) For a pair of matched 2-D and 3-D trihedral angle configurations, we call the trihedral angle constraint is *correctly matched* if the 2-D trihedral angle configuration is indeed the projection of the 3-D trihedral angle configuration, and each pair of 2-D and 3-D sides of the trihedral angle configurations is matched in real corresponding relationship; otherwise, we say the trihedral angle constraint is *an error match.* In real application situation, a obtained trihedral angle constraint may or may not be correctly matched. Therefore, in this section, our second task is to inspect when a correctly matched trihedral angle constraint is derived, if the real solution can be gained by our method; or when an error match is presented, whether our method can identify the ill-condition.

3) It is inevitable that the 2-D data abstracted from a real digital image are affected by noise. To understand the power of our method, our third task in this section is to study the presented approach for its sensitivity to noise.

To make the three questions be tested in general, we arranged our experimental procedure as below.

TABLE II
THE SOLUTION DISTRIBUTION OF (15)

Frequency		Number of Solutions					
		0	1	2	3	4	5
Ideal Data	Correct Match	0	3	48	42	7	0
	Error Match	0	13	58	29	0	0
Noisy Data	Correct Match	0	2	58	39	1	0
	Error Match	0	11	65	24	0	0

TABLE III
RESERVED SOLUTION NUMBER

Frequency		Reserved Solution					
		0	1	2	3	4	5
Ideal Data	Correct Match	0	45	49	5	1	0
	Error Match	85	10	5	0	0	0
Noisy Data	Correct Match	2	52	44	2	0	0
	Error Match	86	8	6	0	0	0

Data-1: Randomly generate a set of ideal trihedral angle constraints in a camera coordinate system. This is a group of ideal data.

Test-1: Use correct angle matching relationship on the ideal data to solve a trihedral angle constraint and then a investigate the solution pattern.

Test-2: Use incorrect angle matching relationship on the ideal data to solve a trihedral angle constraint and then to check the solution results.

Data-2: For a trihedral angle constraint, the effects of different noises can be simply considered as a composite noise acted on the β_1, β_2 and β_3 of (7). Therefore, we choose a noise interval $[-dg, dg]$, for example $[-8, 8]$ with degree measure, as the source of noise. A sequence of noise triplet is randomly selected from the noise interval. Then, each trihedral angle constraint in **Data-1** is added on a noise triplet to product a set of noise data.

Data-3: Do Test-1 for Data-2.

Data-4: Do Test-2 for Data-2.

The test results are given in the following paragraphs.

B. The Initial Solution for Trihedral Angle Constraint

Our solution of a trihedral angle constraint is obtained from the fifth-order equation (15). The equation can be easily solved by iterative approaches. However, we consider the equation (15) as an equation about $\cos\gamma_1$ so only the solutions in the interval $[-1, 1]$ are what we look for. Consequently, we can expect that, in the interval $[-1, 1]$, the number of solutions of equation (15) may be 0, 1, 2, 3, 4 or 5 for a trihedral angle constraint. Therefore, we first investigate the situation about the total number of solutions of (15). According to the procedure depicted in Section IV-A, one hundred groups of data are tested and the result is shown as Table II.

In Table II, an entry represents the emerging frequency of the test case specified by the corresponding row title and column title. For example, the entry 48 in the first row and the third column means that, when using a randomly generated set of an ideal trihedral angle constraint and supposing that the correct match for the constraint has been employed, we got the 2 solution cases for 48 times in the 100 experiments. By Table II, we see that (15) usually has solution in the interval $[-1, 1]$ no matter what kind of experimental condition is assumed. Furthermore, there is no significant difference to distinguish the ideal data from noise data or distinguish the correct match from error match by just referring to solution of (15). In order words, intuitively, we can not find a notable disparity among the distributions of the entries of the four rows in Table II.

C. The Selection of Real Solution for Trihedral Angle Constraint

Once $\cos\gamma_1$ is solved, $\cos\gamma_2$ and $\cos\gamma_3$ can be obtained by (14) and (9). If a set of formal solutions of (15), (14), and (9) is a real solution of a trihedral angle constraint, the solutions must satisfy the inherent criteria **C-1** and **C-2**. We call a this kind of solution set as a reserved solution for a trihedral angle constraint. In other word, a reserved is a real solution of a trihedral angle constraint. Our experiment shows that the reserved solutions have a very different distribution comparing with Table II. Table III is the result.

By Table III, we see that the overwhelming majority of the error matched trihedral angle constraints have no solution. That means that they can be effectively identified by our method. On the other hand, in the case of correctly matched trihedral angle constraints, we noticed that the true solution is always included in the reserved solutions for ideal data; and an approximate solution for the true value always exists in the reserved solutions for noised data (see Section IV-E for case studies). Therefore, our method is well behaved in dealing with real application problem.

Note that in Table II and Table III, we identify a pair of mirror solutions as one solution. In practice situation, if more information and knowledge about the observed object are available, usually the criterion **C-3** and the constraint of visibility can be applied to solve the mirror solution uncertainty.

D. Noise Sensitivity Analysis

We employ the statistical method of regression analysis to explore our technique for its sensitivity in a noise environment.

As we mentioned in Section IV-A, for a trihedral angle constraint (7), the composite effect of noise can be represented by a disturbance on the 2-D angles β_1, β_2 and β_3. For $i = 1, 2, 3$, denote $\hat{\beta}_i$ as the noised β_i; also denote γ_i as the correct solution of (7) corresponding to β_i and $\hat{\gamma}_i$ as the solution of (7) corresponding to $\hat{\beta}_i$. Then, we investigate the covariant relationships for two kinds of corresponding values $(\Delta\gamma_i, \Delta\beta_i)$ and $(\hat{\gamma}_i, \gamma_i)$ by following two linear regression models:

$$\Delta\gamma_i = a_{i0} + a_{i1}\Delta\beta_i + \varepsilon_i \qquad \hat{\gamma}_i = a'_{i0} + a'_{i1}\gamma_i + \varepsilon_i \quad (23)$$

where $\Delta\gamma_i = \hat{\gamma}_i - \gamma_i, \Delta\beta_i = \hat{\beta}_i - \beta_i, (i = 1, 2, 3)$.

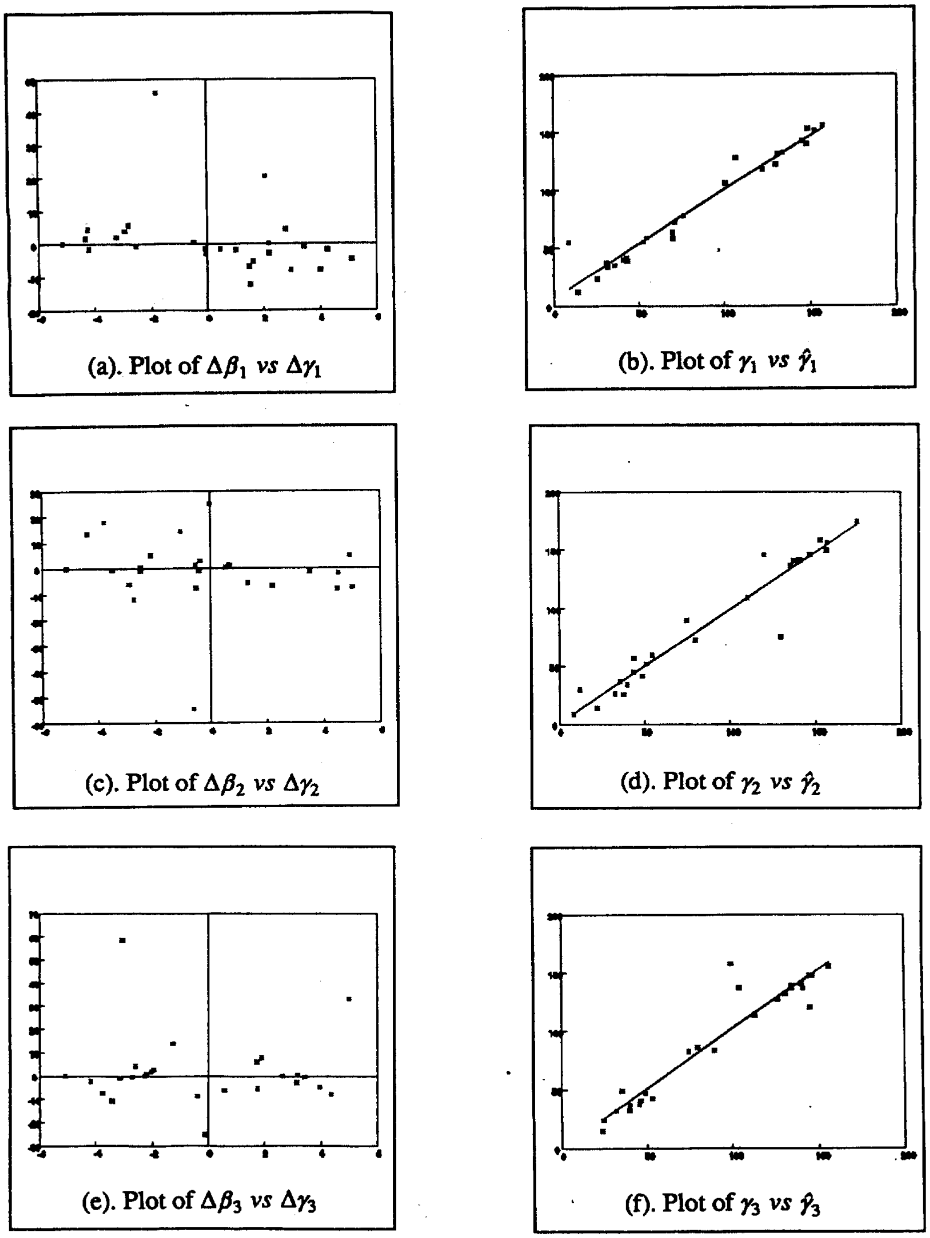


Fig. 7. The plot charts for regression analysis of the noise sensitivity.

According to the procedure described in Section IV-A, twenty five groups of synthetic data were generated for regression analysis; where, the noises were selected from the noise interval $[-5, 5]$ with degree measure; and for multiple solution cases, we chose the best approximation of the correct value γ_i as $\hat{\gamma}_i$. Our intention is to test the null statistical hypotheses:

$$H_0\colon a_i = 0 \text{ and } H_0\colon a_i' = 0 \qquad (i = 1, 2, 3) \tag{24}$$

by using the analysis of variance (ANOVA) to check the data fitness for the linear regression model (23).

The results of the regression analysis are presented by Table IV and Fig. 7. The results shows us that there is no definite relationship between $\Delta\gamma_i$ and $\Delta\beta_i$; but very strong linear relationship exist between $\hat{\gamma}_i$ and γ_i. Therefore, the solution of our method for trihedral angle constraint is stable under the noise environment with $[-5°, 5°]$ noise interval. More generally, we can expect that the similar results will occur for different but reasonable noise intervals. In fact, this is true for our another test with noise interval $[-8°, 8°]$. We choose $[-5°, 5°]$ as our noise interval because experimentally we consider the interval can cover noise range in normal situation.

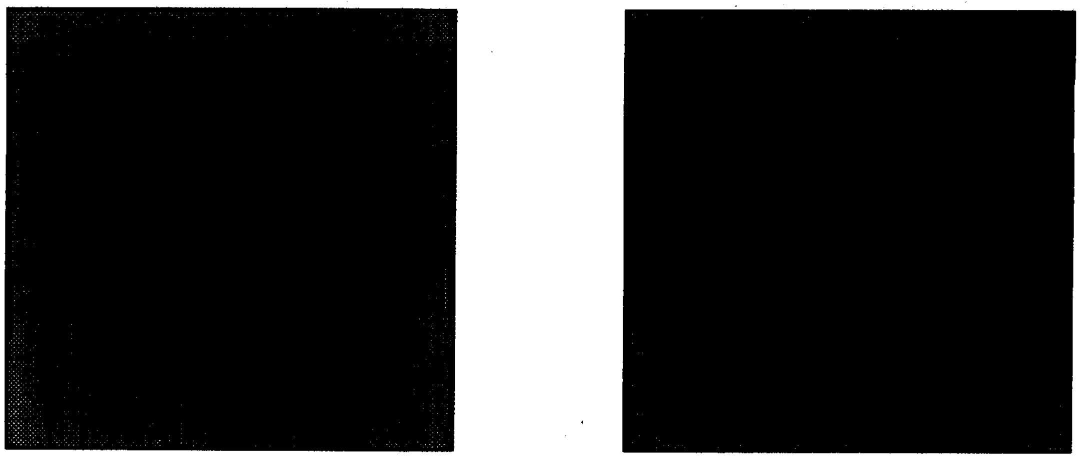


Fig. 8. The solution configurations of case 1. (a) The ideal solution. (b) The noised solution.

TABLE IV
HYPOTHESIS TEST FOR THE REGRESSION ANALYSIS

Models	F_value for the Null Hypotheses	P_value for the Null Hypotheses	Acceptance for the Null Hypotheses
$\Delta\gamma_1 = a_{10} + a_{11}\Delta\beta_1$	1.811	0.1910	Accept
$\Delta\gamma_2 = a_{20} + a_{21}\Delta\beta_2$	0.532	0.4728	Accept
$\Delta\gamma_3 = a_{30} + a_{31}\Delta\beta_3$	0.001	0.9821	Accept
$\hat{\gamma}_1 = a'_{10} + a'_{11}\gamma_1$	464.508	0.0001	Reject
$\hat{\gamma}_2 = a'_{20} + a'_{21}\gamma_2$	353.901	0.0001	Reject
$\hat{\gamma}_3 = a'_{30} + a'_{31}\gamma_3$	253.391	0.0001	Reject

TABLE V
EXAMPLE OF SINGLE SOLUTION CASE

Trihedral Angle Constraint Configuration					
η_{12}	η_{13}	η_{23}	67.571604	86.834868	69.342293
γ_1	γ_2	γ_3	130.853547	134.728754	138.439739
β_1	β_2	β_3	88.523299	- 9.910554	- 121.549899
$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	86.838034	- 8.336878	- 123.994752
Solution Pattern (* 's denote the abandoned extraneous solutions)					
$\cos\gamma_1$	$\cos\gamma_2$	$\cos\gamma_3$	γ_1	γ_2	γ_3
0.970826	- 0.053371	2.835116	*	*	*
0.999666	0.385158	0.032857	*	*	*
- 0.654128	- 0.703751	- 0.748258	130.853547	134.728754	138.439739
$\cos\hat{\gamma}_1$	$\cos\hat{\gamma}_2$	$\cos\hat{\gamma}_3$	$\hat{\gamma}_1$	$\hat{\gamma}_2$	$\hat{\gamma}_3$
0.801787	- 0.059474	3.339292	*	*	*
0.983277	0.409125	- 0.098713	*	*	*
- 0.626433	- 0.717331	- 0.763436	128.787414	135.834538	139.768061

For these reasons, we conclude that the method is robust in real application situation.

E. The Case Studies

In this section, different solution patterns for trihedral angle constraint will be illustrated in detail. For each case, first, the ideal image data $\beta_1, \beta_2, \beta_3$ and the noised image data $\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3$ are produced depending on a trihedral angle which is specified by the three angles of η_{12}, η_{13} and η_{23}; then, the

TABLE VI
EXAMPLE OF TWO SOLUTION CASE

Trihedral Angle Constraint Configuration					
η_{12}	η_{13}	η_{23}	35.843159	53.146751	40.396609
γ_1	γ_2	γ_3	142.100803	110.233148	117.820162
β_1	β_2	β_3	125.522986	146.621923	- 169.698540
$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	121.557323	147.422091	- 165.707698
Solution Pattern (* 's denote the abandoned extraneous solutions)					
$\cos\gamma_1$	$\cos\gamma_2$	$\cos\gamma_3$	γ_1	γ_2	γ_3
0.693679	0.098478	0.360203	*	*	*
0.789093	0.345841	0.466698	37.899191	69.766851	62.179838
- 0.534381	- 0.912160	- 0.567319	122.301938	155.805602	124.563475
$\cos\hat{\gamma}_1$	$\cos\hat{\gamma}_2$	$\cos\hat{\gamma}_3$	$\hat{\gamma}_1$	$\hat{\gamma}_2$	$\hat{\gamma}_3$
0.775503	0.149247	0.452547	*	*	*
0.855906	0.466091	0.551110	31.139994	62.219150	56.556799
- 0.661338	- 0.967446	- 0.651532	131.402019	165.340355	130.657243

derived solutions $\gamma_1, \gamma_2, \gamma_3$ and $\hat{\gamma}_1, \hat{\gamma}_2, \hat{\gamma}_3$ will be shown by tables and displayed by wireframe pictures.

Case 1) Single Solution Case: For a trihedral angle constraint, a single solution is mostly encountered (see Table III). An example of a single solution case is shown in Table V, and the two solutions are displayed in Fig. 8. We see in the Table V, as well as in the following tables for the case 2 to 4, a solution which matches the original data always can be obtained in the solutions of the ideal data. Also, in most cases, it is actually difficult to tell the difference of a pair of corresponding ideal and noise solutions by watching the solution figures.

Case 2) Two Solution Case: In the example of the two solutions case shown in Table VI, the first solution derived from ideal data is the one which matches the original trihedral angle configuration but as a mirror image. The other solution derived from ideal data is an approximate solution to the original trihedral angle configuration. In fact, we have found that when there are multiple solutions in a trihedral angle constraint, usually these solutions are spread around the two correct mirror solutions respectively in some degree. This property may be utilized to classify the multiple

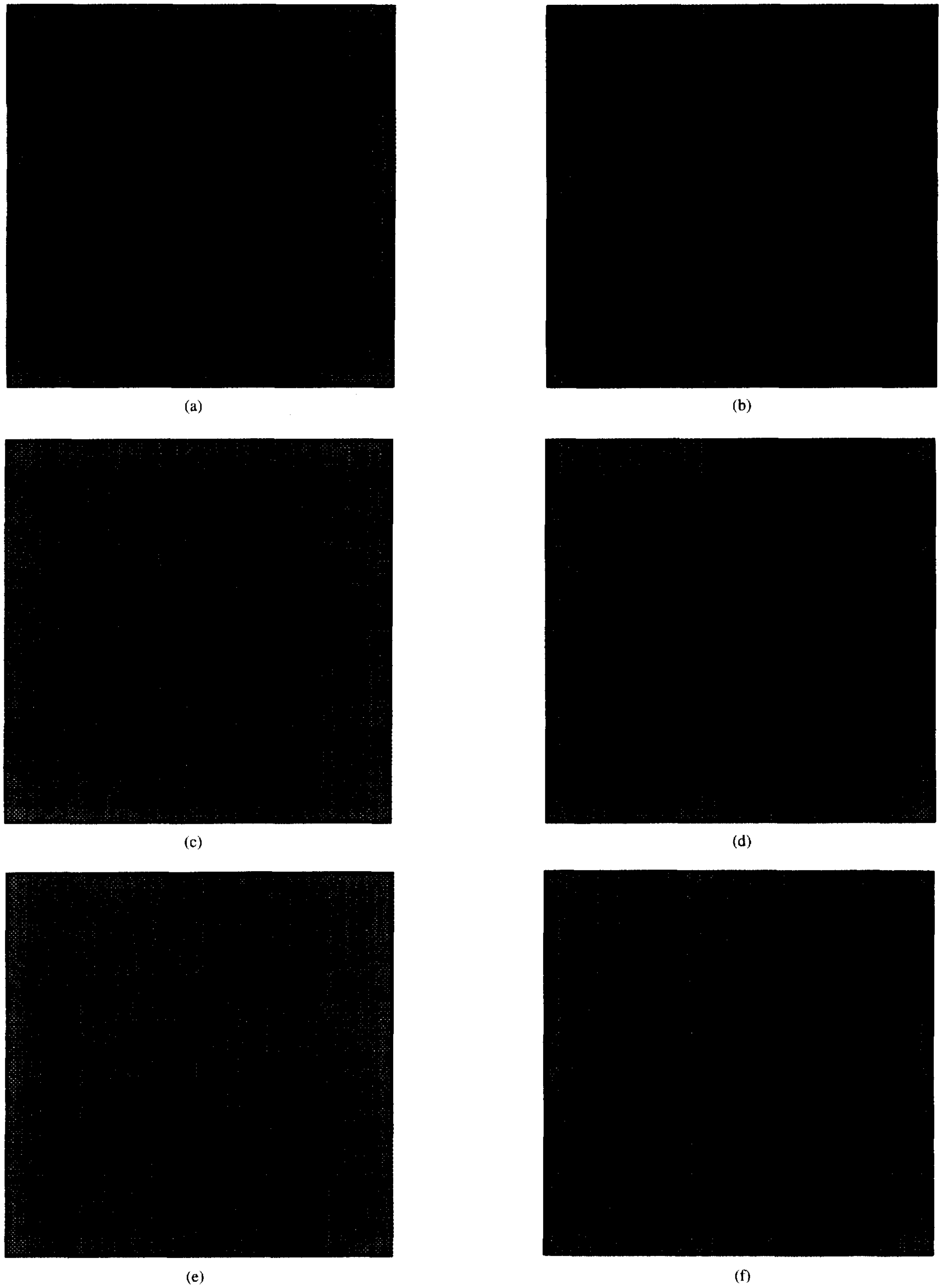


Fig. 9. The solution configurations of case 2. (a) The first ideal solution. (b) The first noised solution. (c) The second ideal solution. (d) The second noised solution. (e) Watch the second ideal solution from another view position. (f) Watch the second noised solution from another view position.

solutions into two groups for further processing in application. The solutions in Table VI are displayed in Fig. 9. Fig. 9(a)–(d) are the four solutions observed from a same view position; where, the side formed by P_0 and P_3 in Fig. 9(c) or (d) is occluded by the face formed by P_0, P_1 and P_2. Fig. 9(e) and (f) are the pictures of watching the two

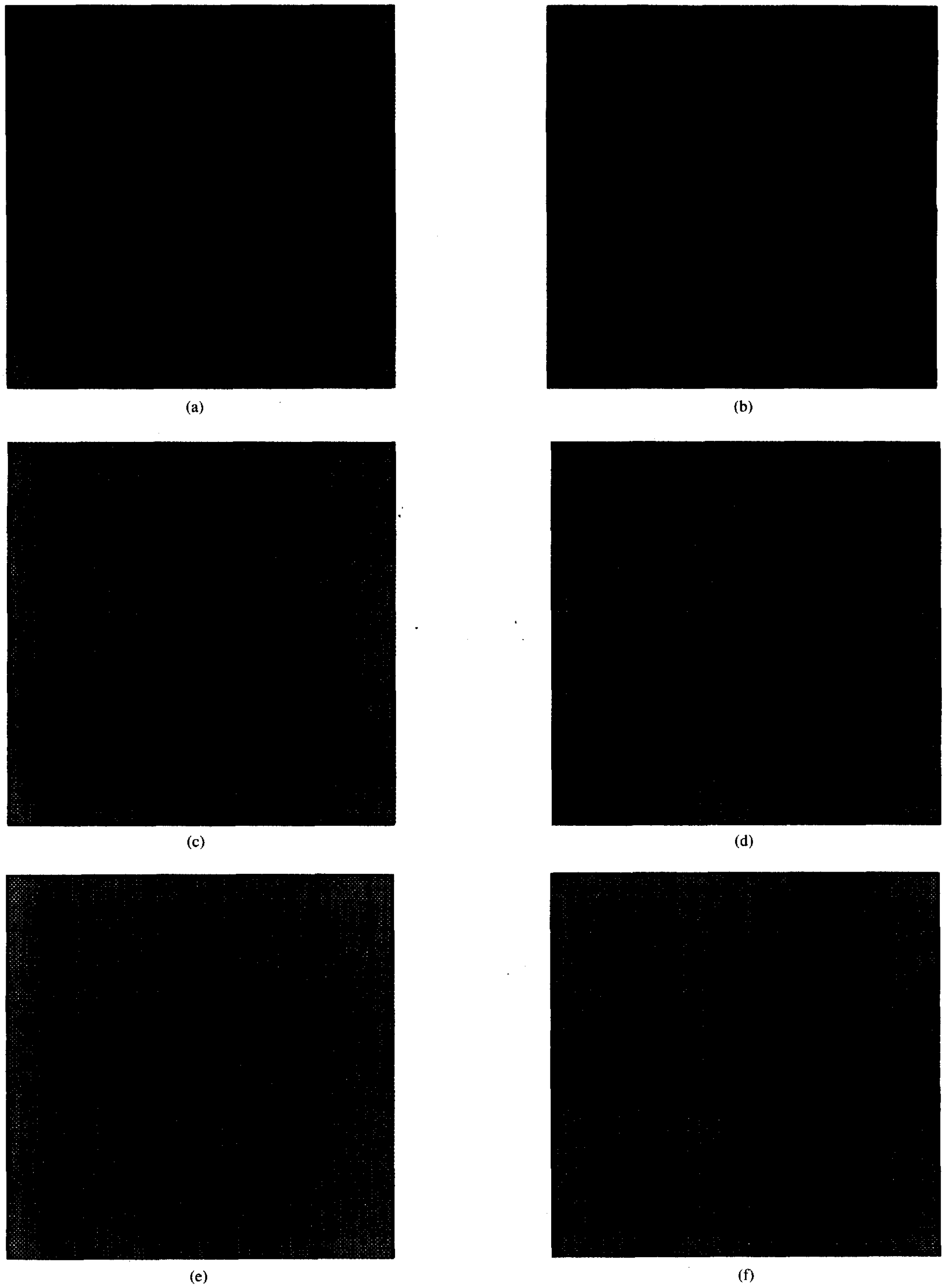
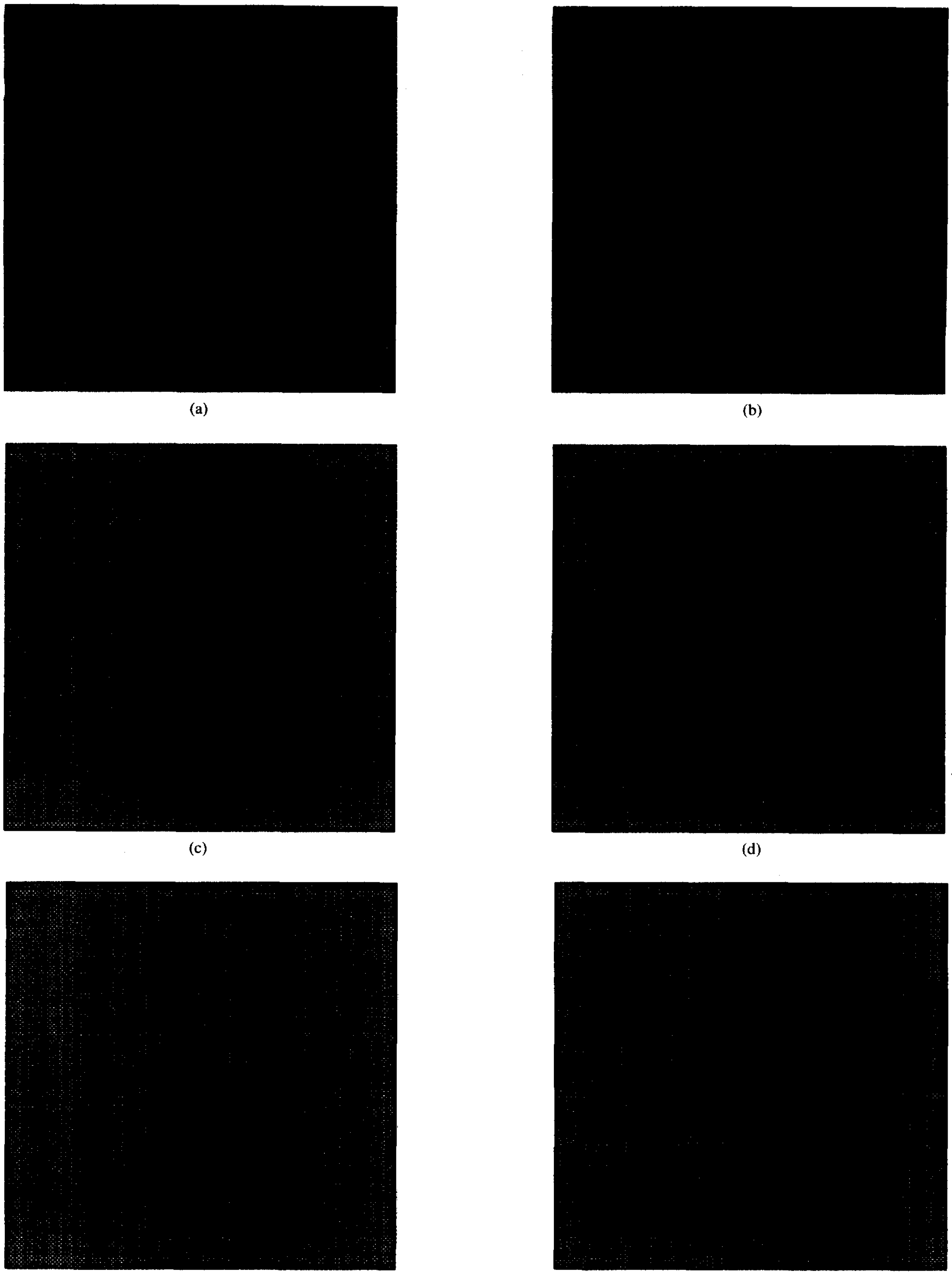


Fig. 10. The solution configurations of case 3. (a) The first ideal solution. (b) The first noised solution. (c) The second ideal solution. (d) The second noised solution. (e) The third ideal solution. (f) The third noised solution.

solutions displayed in Fig. 9(c) and (d), but from another view position. This time, they look like the picture Fig. 9(a) and (b).

Case 3) Three Solution Case: We show two examples for the three solutions case and the four solutions case in the following text. However, actually the situation that the number



(a) (b) (c) (d) (e) (f)

Fig. 11. The solution configurations of case 4. (a) The first ideal solution. (b) The first noised solution. (c) The second ideal solution. (d) The second noised solution. (e) The third ideal solution. (f) The fourth ideal solution.

of solutions is more than two is very few for trihedral angle constraint. Among the several hundreds of trihedral angle constraints we generated randomly, the case of three solutions or four solutions did not exceed 15, and we have not found a case of five solutions although the solutions is derived from the fifth-order (15).

TABLE VII
EXAMPLE OF THREE SOLUTION CASE

Trihedral Angle Constraint Configuration					
η_{12}	η_{13}	η_{23}	58.439898	65.877503	95.646584
γ_1	γ_2	γ_3	84.507202	100.310751	131.562371
β_1	β_2	β_3	- 162.343584	- 105.843221	147.001394
$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	- 164.965500	-108.952804	152.079960
Solution Pattern (* 's denote the abandoned extraneous solutions)					
$\cos\gamma_1$	$\cos\gamma_2$	$\cos\gamma_3$	γ_1	γ_2	γ_3
0.270464	- 0.028656	0.970935	74.308129	91.642089	13.847847
0.605293	0.985947	- 0.149213	*	*	*
- 0.095721	0.178987	0.663435	95.492798	79.689249	48.437629
- 0.989730	- 0.457956	- 0.326374	171.781300	117.255304	109.048828
$\cos\hat{\gamma}_1$	$\cos\hat{\gamma}_2$	$\cos\hat{\gamma}_3$	$\hat{\gamma}_1$	$\hat{\gamma}_2$	$\hat{\gamma}_3$
0.338215	- 0.081971	0.994059	70.231821	94.701892	6.248383
0.681931	0.954819	- 0.151006	*	*	*
- 0.346582	0.075637	0.490185	110.278410	85.662193	60.646586
- 0.918088	- 0.330667	- 0.142751	156.648091	109.309237	98.207079

TABLE VIII
EXAMPLE OF FOUR SOLUTION CASE

Trihedral Angle Constraint Configuration					
η_{12}	η_{13}	η_{23}	146.871170	79.592041	129.182774
γ_1	γ_2	γ_3	96.639703	78.120769	128.700834
β_1	β_2	β_3	- 87.713150	125.455557	- 5.748628
$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	-92.570143	121.797076	- 3.414552
Solution Pattern					
$\cos\gamma_1$	$\cos\gamma_2$	$\cos\gamma_3$	γ_1	γ_2	γ_3
0.313425	- 0.544181	0.158202	71.734242	122.968708	80.897424
- 0.115625	0.205850	- 0.625254	96.639703	78.120767	128.700839
- 0.133183	0.077661	- 0.405648	97.653541	85.545866	113.931743
- 0.768348	0.477150	- 0.119515	140.205757	61.500589	96.864128
$\cos\hat{\gamma}_1$	$\cos\hat{\gamma}_2$	$\cos\hat{\gamma}_3$	$\hat{\gamma}_1$	$\hat{\gamma}_2$	$\hat{\gamma}_3$
0.463715	- 0.732774	0.363351	62.372918	137.119448	68.693866
- 0.256301	0.340939	- 0.663264	104.850682	70.065884	131.549258

In the example of the three solutions case shown in Table VII, the correct ideal solution is the second one. Again, the correct solution is the mirror image to the original configuration. Fig. 10 shows the pictures of the solutions in Table VII.

Case 4) Four Solution Case: In the example of the four solutions case shown in Table VIII, we can notice that the number of solutions is four for ideal data but just two for noisy data. The situation that the number of solutions for noisy data is less than that for ideal data is very common for trihedral angle constraint. By comparing Table II and Table III, we can have a knowledge for this situation. On the other hand, although the number of solutions for noisy data may be less than that for ideal data, we notice that the approximation of the correct solution can be obtained by noisy solution in the overwhelming majority cases.

The pictures of the six solutions in Table VIII are displayed in Fig. 11.

V. CONCLUSION

Methods for solving the orientation and position of an object from a single perspective projection view are important for their wide applications and powers. The method presented in this paper permits us to find an analytic solution of a trihedral angle constraint by directly using angle information. Angle is a very common feature for characterizing a variety of objects. The knowledge about the angles of an object provides a strong clue for estimating the orientation and position of the object. So the constraints involving angles have been studied and applied in computer vision and image analysis by many researchers. Our method gives the first closed form solution for the problem of angle constraint in perspective projection. Trihedral angle is the simplest but also the most encountered angle constraint in 3-D computer vision. For different cases of trihedral angle constraint depicted in Fig. 1, the proposed approach can be effectively used to recover the orientation and position of an object. Furthermore, our method also provides a basic approach for dealing with the general PnA problems provided that the number of constraint equations on PnA problem is greater than or equal to the number of unknowns. The results of simulation experiments show that the new method is not only a real time technique of shape from angle constraint, but also powerful enough to cope with noisy environments in real applications. With the new developments, we present a overall analysis on the essential characteristics of PnP, PnL, and PnA, the three fundamental techniques used for the problem of shape from inverse perspective projection. The combination of the three techniques of PnP, PnL, and PnA certainly is a very promising tool to deal with various situations in the problem of shape from perspective. To design a sound algorithm for this unified approach is a topic for our further research.

REFERENCES

[1] R. M. Haralick, "Monocular vision using inverse perspective projection geometry: analytic relations," in *Proc. IEEE Conf. CVPR*, 1889, pp. 370–378.
[2] P. G. Mulgaonkar, L. G. Shapiro, and R. M. Haralikc, "Shape from perspective: A rule based approach," *CVGIP*, vol. 36, pp. 298–320, 1989.
[3] K. Kanatani, "Constraints on length and angle," *CVGIP*, vol. 41 pp. 28–42, 1988.
[4] S. T. Barnard, "Choosing a basis for perceptual space," *CVGIP*, vol. 29, pp. 87–99, 1985.
[5] T. Shakunaga and H. Kaneko, "Shape from angles under perspective projection," in *Proc. IEEE 2nd Int. Conf. on CV*, 1988, pp. 671–678.
[6] T. Kanade, "Recovery of the three-dimensional shape of an object from a single view," *Artificial Intell. J.*, vol. 17, pp. 409–460, 1981.
[7] R. Horaud, "New method for matching 3-D object with single perspective views," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-9, no. 3, pp. 401–412, 1987.
[8] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with application to image analysis and automated cartography," *Commun. ACM.*, vol. 24, no. 6, pp. 381–395, 1981.
[9] S. Linnainma, D. Harwood, and L. S. Davis, "Pose determination of a three-dimensional object using triangle pairs," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, no. 5, pp. 634–646, 1988.
[10] D. G. Lowe, "Three-dimensional object recognition from single two-dimensional image," *Artificial Intell. J.*, vol. 13, pp. 355–395, 1987.
[11] R. Horaud, B. Conio, and O. Leboulleux, "An analytic solution for the perspective 4-point problem," *Comput. Vision Graphics and Image Processing*, vol. 47, pp. 33–44, 1989.
[12] M. Dhome, M. Richetin, J.-T. Lapreste, and G. Rives, "Determination of the attitude of 3-D objects from a single perspective view," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 1265–1278, 1989.
[13] H. H. Chen, "Pose determination from line-to-plane correspondences: existence condition and closed-from solution," in *Proc. IEEE 3rd Int. Conf. Comput. Vision*, 1990, pp. 374–378.
[14] R. M. Haralick, H. Joo, C. N. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kin, "Pose Estimation from correspoding point data," *IEEE Trans. Syst., Man Cybern.*, vol. 19, no. 6, pp. 1426–1446, 1989.
[15] K. Kanatani, *Geometric Computation for Machine Vision.* Oxford: Oxford Univ. Press, 1993.
[16] Y. Wu, S. S. Iyengar, R. Jain, and S. Bose, "Shape from perspective trihedral angle constraint," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 1993, pp. 261–266.

Yuyan Wu has been a graduate student in the Department of Computer Science at LSU since June 1991. Mr. Wu finished his Master's degree in computer science at LSU in December 1993.

His research interests are in the area of computer vision, image understanding and general purpose vision systems.

S. Sitharama Iyengar (SM'89) received the Ph.D. degree in 1974.

He has directed over 18 Ph.D. dissertations at LSU. He has served as principle investigator on research projects supported by the Office of the Naval Research, the National Aeronautics and Space Administration, the National Science Foundation/Laser Program, the California Instititute of Technology's Jet Propulsion Laboratory, the Department of Navy-NORDA, the Department of Energy (through Oak Ridge National Laboratory, Tennesse), the LEQFS-Board of Regents, and the Apple Computers. He is Chairman of the Computer Science Department and Professor of Computer Science at Louisiana State University. He has directed LSU's Robotics Research Laboratory since its inception in 1986. He has been actively involved with research in high performance algorithms and data structures.

In addition to a two-volume tutorial, *Autonomous Robots* (IEEE Press), he has edited two other books and over 150 publications— including 85 archival journal papers in areas of high-performance parallel and distributed algorithms and data structure for image processing and pattern recognition, autonomous navigation, and distributed sensor networks. Dr. Iyengar was a Visiting Professor (fellow) at JPL, the Oak Rigde National Laboratory, and the Indian Institute of Science. He is also an Association of Computing Machinery national lecturer, a series editor for *Neuro Computing of Complex Systems*, and area editor for the *Journal of Computer Science and Information*. He has served as guest editor for the IEEE TRANSACTIONS ON SOFTWARE ENGINEERING (1988); *Computer Magazine* (1989); the IEEE TRANSACTIONS ON SYSTEM, MAN, AND CYBERNATICS; the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING; the *Journal of Computers and Electrical Engineering*; and the *Jouranl of Theoritical Comptuer Science*. He was awarded the Phi Delta Kappa Research Award of Distinction at LSU in 1989, won the Best Teacher Award in 1978, and received the WIlliams Evans Fellowship from the University of Otago, New Zealand, in 1991.

Ramesh Jain (F'92) received the B.E. degree from Nagpur University in 1969 and the Ph.D. degree from IIT, Kharagpur in 1975.

He is currently a Professor of Electrical and Computer Engineering, and Computer Science and Engineering at University of California at San Diego. Before joining UCSD, he was a Professor of Electrical Engineering and Computer Science, and the founding Director of the Artificial Intelligence Laboratory at the University of Michigan, Ann Arbor, MI. He has also been affiliated with Stanford University, IBM Almaden Research Labs, General Motors Research Labs, Wayne State University, University of Texas at Austin, University of Hamburg, West Germany, and Indian Institute of Technology, Kharagpur, India. His current research interests are in multimedia information systems, image databases, machine vision, and intelligent systems. He is the Chairman of Imageware Inc., an Ann Arbor based company dedicated to revolutionize software interfaces for emerging sensor technologies.

Dr. Ramesh is a Fellow of AAAI, and Society of Photo-Optical Instrumentation Engineers, and member of ACM, Pattern Recognition Society, Manufacturing Engineers. He has been involved in organization of several professional conferences and workshops, and served on editorial boards of many journals. Currently, he is the Editor-in-Chief of *IEEE Multimedia*, and is on the editorial boards of *Machine Vision and Applications*, *Pattern Recognition*, and *Image and Vision Computing*.

Santanu Bose has been a graduate student in the Department of Computer Science at LSU since June 1991. Mr. Bose finished his Master's degree in computer science in the area of vision at LSU in summer of 1993.

His research interests are in the area of computer vision and pattern recognition.