Pose Estimation from Corresponding Point Data

Robert M. Haralick, Chung Nan Lee, Xinhua Zhuang, Vinay G. Vaidya, and Man Bae Kim

Department of Electrical Engineering
University of Washington

Abstract

Pose estimation is an essential step in many machine vision problems involving the estimation of object position and orientation relative to a model reference frame or relative to the object position and orientation at a previous time using a camera sensor or a range sensor.

Solutions for three different pose estimation problems are presented. Closed form least squares solutions are given to the over constrained 2D-2D and 3D–3D pose estimation problems. A simplified linear solution to the 2D perspective projection–2D perspective projection pose estimation problem is also given.

Simulation experiments consisting of hundreds of thousands of trials having varying numbers of pairs of corresponding points, varying signal to noise ratio with either Gaussian or Uniform noise provide data suggesting that accurate inference of rotation and translation with noisy data may require corresponding point data sets having hundreds of corresponding point pairs when the signal to noise ratio is less than 40 db.

1. Introduction

There are four pose estimation problems with point data. Each arises from two views taken of the same object which can be thought of as having undergone an unknown rigid body motion from the first view to the second view. In model based vision, one "view" provides data relative to the model reference frame. In motion estimation and structure from motion problems there is a rigid body motion of the sensor, the object or both. In any case, in each problem corresponding point pairs from the two views are obtained from some kind of matching procedure. The pose estimation problem with corresponding point data begins with such a corresponding point data set. Its solution is a procedure which uses the corresponding point data set to estimate the translation and rotation which define the relationship between the two coordinate frames.

In the simplest pose estimation problem, the data sets consist of two-dimensional data points in a two-dimensional space. Such data sets arise naturally when flat 3D objects are viewed under perspective projection with the look angle being the same as the surface normal of the object viewed. In the next more difficult pose estimation problem, the data sets consist of three-dimensional data points in a three-dimensional space. Such data sets arise naturally when 3D objects are viewed with a range finder sensor. In the most difficult pose estimation problems, one data set consists of 2D perspective projection of 3D points and the other data set consists of either a 3D point data set, in which case it is known as the camera calibration problem, or the other data set consists of a second 2D perspective projection view of the same 3D point data set. The latter case occurs with time-varying imagery.

Section 2 derives a closed form least squares solution to the pure 2D-2D pose estimation problem. Section 3 derives a closed form least squares solution to the pure 3D-3D pose estimation problem using a singular value decomposition technique. The least squares solution for both the 2D–2D and 3D–3D pose estimation problems are constrained to produce rotation matrices which are guaranteed to be orthonormal. Section 4 discusses a solution to the 2D perspective projection–2D perspective projection pose estimation problem. Sections 3 and 4 contain experimental results showing the effect of noise and number of data points on the results.

2. 2D–2D Estimation

There are a variety of model based inspection tasks which require the coordinate system of an object model to be aligned with the coordinate system of a set of observations before the actual inspection judgements can be made. One example is surface mount device inspection on printed circuit boards. Here, the image processing produces, among other measurements, the observed center position of each device. The model stores, in the printed circuit board coordinate system, the center positions, orientations, and sizes of all devices. To determine whether each device which should be present is present, and whether everything observed to be present is actually present and in its correct position and orientation first requires determining the relationship between the coordinate system of the observed image and the coordinate system of the model. Usually this relationship is given by a two-dimensional rotation and translation.

TH0210-5/87/0000/0258$01.00 © 1987 IEEE

Section 2.1 gives a precise statement of this problem as a weighted least squares problem. Section 2.2 is a derivation of the solution to the weighted least squares problem.

2.1 Statement of Problem

In the simple two-dimensional pose detection problem, we are given N two-dimensional coordinate observations from the observed image: $x_1, ..., x_N$. These could correspond, for example, to the observed center position of all observed objects. We are also given the corresponding or matching N two-dimensional coordinate vectors from the model: $y_1, ..., y_N$. In the usual inspection situation, establishing which observed vector corresponds to which model vector is simple because the object being observed is fixtured and its approximate position and orientation is known. The approximate rotational and translational relationship between the image coordinate system and the object coordinate system permits the matching to be done just by matching a rotated and translated image position to an object position. The match is established if the rotated image position is close enough to the object position.

In the ideal case, the simple two-dimensional pose detection problem is to determine from the matched points a more precise estimate of a rotation matrix R and a translation t such that $y_n = Rx_n + t$, $n = 1, ..., N$. Since there are likely to be small observational errors, the real problem must be posed as a minimization. Determine R and t which minimize the weighted sum of the residual errors $\epsilon^2 = \sum_{n=1}^{N} w_n \|y_n - (Rx_n + t)\|^2$. The weights w_n, $n = 1, ..., N$ satisfy $w_n \geq 0$ and $\sum_{n=1}^{N} w_n = 1$. If there is no prior knowledge as to how the weights should be set, they can be defined to be equal: $w_n = 1/N$.

2.2 Derivation

Upon expanding the previous equation and using the fact that $R^{-1} = R'$, there results $\epsilon^2 = \sum_{n=1}^{N} w_n \left[(y_n - t)'(y_n - t) - 2(y_n - t)'Rx_n + x_n'x_n\right]$ Taking the partial derivative of ϵ^2 with respect to the components of the translation t and setting the partial derivative to 0 and letting $\bar{x} = \sum_{n=1}^{N} w_n x_n$ and $\bar{y} = \sum_{n=1}^{N} w_n y_n$ there immediately results $\bar{y} = R\bar{x} + t$.

Substituting $\bar{y} - R\bar{x}$ for t in the expression for the residual error we can do some simplifying and we obtain $\epsilon = \sum_{n=1}^{N} w_n \left[(y_n - \bar{y})'(y_n - \bar{y}) - 2(y_n - \bar{y})'R(x_n - \bar{x}) + (x_n - \bar{x})'(x_n - \bar{x})\right]$.

The counterclockwise rotation angle θ is related to the rotation matrix by

$$R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

We want to take the partial derivative of ϵ^2 with respect to θ. Now we need a notation in which the two components of x_n and the two components of y_n can be written explicitly. Letting

$$x_n = \begin{pmatrix} x_{1n} \\ x_{2n} \end{pmatrix}, \quad y_n = \begin{pmatrix} y_{1n} \\ y_{2n} \end{pmatrix}$$

$$\bar{x} = \begin{pmatrix} \bar{x}_1 \\ \bar{x}_2 \end{pmatrix}, \text{ and } \bar{y} = \begin{pmatrix} \bar{y}_1 \\ \bar{y}_2 \end{pmatrix}$$

then $(y_n - \bar{y})'R(x_n - \bar{x}) = (y_{n1} - \bar{y}_1)\cos\theta(x_{n1} - \bar{x}_1) + (y_{n1} - \bar{y}_1)(-\sin\theta)(x_{n2} - \bar{x}_2) + (y_{n2} - \bar{y}_2)\sin\theta(x_{n1} - \bar{x}_1) + (y_{n2} - \bar{y}_2)\cos\theta(x_{n2} - \bar{x}_2)$ Then, setting to zero the partial derivative of ϵ^2 with respect to θ results in $0 = -2\sum_{n=1}^{N} w_n \Big[(y_{n1} - \bar{y}_1)(-\sin\theta)(x_{n1} - \bar{x}_1) + (y_{n1} - \bar{y}_1)(-\cos\theta)(x_{n2} - \bar{x}_2) + (y_{n2} - \bar{y}_2)\cos\theta(x_{n1} - \bar{x}_1) + (y_{n2} - \bar{y}_2)(-\sin\theta)(x_{n2} - \bar{x}_2)\Big]$. Letting

$$A = \sum_{n=1}^{N} w_n \left[(y_{n1} - \bar{y}_1)(x_{n1} - \bar{x}_1) + (y_{n2} - \bar{y}_2)(x_{n2} - \bar{x}_2)\right]$$

$$B = \sum_{n=1}^{N} w_n \left[(y_{n1} - \bar{y}_1)(x_{n2} - \bar{x}_2) - (y_{n2} - \bar{y}_2)(x_{n1} - \bar{x}_1)\right]$$

Then $0 = A\sin\theta + B\cos\theta$ Hence, $\cos\theta = \frac{-A}{\sqrt{A^2+B^2}}$ and $\sin\theta = \frac{B}{\sqrt{A^2+B^2}}$ or $\cos\theta = \frac{A}{\sqrt{A^2+B^2}}$ and $\sin\theta = \frac{-B}{\sqrt{A^2+B^2}}$

The correct value for θ will, in general, be unique and will be that θ which minimizes ϵ^2. Thus the better of the two choices can always be easily determined by simply substituting each value for θ into the original expression for ϵ^2.

2.3 Experimental Results

For each trial, object data points were generated uniformly in the square $[-2, 2] \times [-2, 2]$. A rotation angle was chosen from the interval $[-15, 15]$ (in degrees) according to a uniform distribution and the translation vector was chosen from the square $[-1, 1] \times [-1, 1]$ also according to a uniform distribution. Independent Gaussian noise was added to the rotated and translated points and the signal to noise ratio, defined as 20 log peak-to-peak signal/rms noise, was varied between 0 db and 52 db. For each different combination of signal to noise ratio and number of corresponding point pairs, one thousand trials were made.

The mean absolute error of the rotation angle as a function of signal to noise ratio for number of corresponding point pairs varying between 8 and 200 was computed. For number of corresponding point pairs equal to 8, the signal to noise ratio must exceed 40 db to guarantee mean absolute error of less than 1 degree while for 100 corresponding point pairs the signal to noise ratio can go as low as 25 db while maintaining a less than 1 degree mean absolute rotation error.

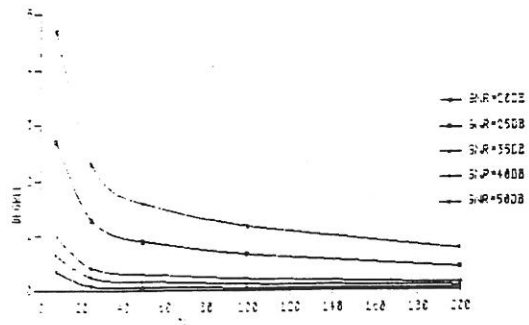

Figure 1. *Mean absolute rotational error as a function of the number of corresponding point pairs for the 2D–2D pose estimation problem.*

The pattern for mean translational distance error is similar. To maintain a mean translational distance error of .01, which is a relative error of about .25%, requires 100 corresponding point pairs at a 32 db signal to noise ratio. Using only 8 corresponding point pairs, even a signal to noise ratio of 52 db provides a mean translation distance error of about .03, or .75%.

Figure 1 shows rotational error as a function of number of corresponding point pairs for a few values of signal to noise ratio. Figure 1 suggests a rapid increase in expected error when there are fewer than 50 corresponding point pairs for a 35 db signal to noise ratio.

3. 3D–3D Estimation

3.1 Statement of Problem

Let $y_1 ... y_N$ be N points in Euclidean 3-space. Let R be a rotation matrix and t be a translation vector. Let $x_1, ..., x_N$ be the points in Euclidean 3-space which match $y_1, ..., y_N$. Each x_n is the same rigid body motion of y_n. Hence each x_n is obtained as a rotation of y_n plus a translation plus noise: $x_n = Ry_n + t + \eta_n$. The simple 3D pose detection problem is to infer R and t from $x_1, ..., x_N$ and $y_1, ..., y_N$.

3.2 Derivation

To determine R and t we set up a constrained least squares problem. We will minimize $\sum_{n=1}^{N} \|x_n - (Ry_n + T)\|^2$ subject to the constraint that R is a rotation matrix, that is, $R' = R^{-1}$. To be able to express these constraints using Lagrangian multipliers we let

$$R = \begin{pmatrix} r_1' \\ r_2' \\ r_3' \end{pmatrix} \text{ where each } r_i \text{ is a } 3 \times 1 \text{ vector}$$

The constraint $R' = R^{-1}$, then amounts to the six constraint equations $r_1'r_1 = 1$, $r_2'r_2 = 1$, $r_3'r_3 = 1$, $r_1'r_2 = 0$, $r_1'r_3 = 0$, $r_2'r_3 = 0$ The least squares problem witb constraints given by these equations can be written as minimizing ϵ^2 where $\epsilon^2 = \sum_{n=1}^{N}\sum_{k=1}^{3}(x_{nk} - r_k'y_n - t_k)^2 + \sum_{k=1}^{3}\lambda_k(r_k'r_k - 1) + 2\lambda_4 r_1'r_2 + 2\lambda_5 r_1'r_3 + 2\lambda_6 r_2'r_3$

$$x_n = \begin{pmatrix} x_{n1} \\ x_{n2} \\ x_{n3} \end{pmatrix}, \quad y_n = \begin{pmatrix} y_{n1} \\ y_{n2} \\ y_{n3} \end{pmatrix}, \text{ and } \quad t = \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix}$$

Taking the partial derivative of ϵ^2 with respect to t_n, and setting these partials to zero results in $\sum_{n=1}^{N}(x_n - Ry_n - t) = 0$ By rearranging we obtain $t = \overline{x} - R\overline{y}$ where $\overline{x} = \frac{1}{N}\sum_{n=1}^{N} x_n$ and $\overline{y} = \frac{1}{N}\sum_{n=1}^{N} y_n$.

Thus once R is known, t is quickly determined. Substituting $\overline{x} - R\overline{y}$ for t in the definition of ϵ^2, there results

$$\epsilon^2 = \sum_{n=1}^{N}\sum_{k=1}^{3}\Big(x_{nk} - \overline{x}_n - r_k'(y_n - \overline{y})\Big)^2 + \sum_{k=1}^{3}\lambda_k(r_k'r_k - 1) + 2\lambda_4 r_1'r_2 + \lambda_5 r_1'r_3 + \lambda_6 r_2'r_3$$

where

$$\overline{x} = \begin{pmatrix} \overline{x}_1 \\ \overline{x}_2 \\ \overline{x}_3 \end{pmatrix}, \quad \overline{y} = \begin{pmatrix} \overline{y}_1 \\ \overline{y}_2 \\ \overline{y}_3 \end{pmatrix}.$$

Now we take partial derivatives of ϵ^2 with respect to the components of each y_n. To write things more compactly, by $\frac{\partial \epsilon^2}{\partial r_n}$ we mean a 3×1 vector whose components are the partial derivative of ϵ^2 with respect to each of the components of r_n. Then,

$$\begin{aligned}
\frac{\partial \epsilon^2}{\partial r_1} &= \sum_{n=1}^{N} 2\Big(x_{n1} - \overline{x}_1 - r_1'(y_n - \overline{y})\Big)(y_n - \overline{y})(-1) \\
&\quad + 2\lambda_1 r_1 + 2\lambda_4 r_2 + 2\lambda_5 r_3 \\
\frac{\partial \epsilon^2}{\partial r_2} &= \sum_{n=1}^{N} 2\Big(x_{n2} - \overline{x}_2 - r_2'(y_n - \overline{y})\Big)\,(y_n - \overline{y})\,(-1) \\
&\quad + 2\lambda_2 r_2 + 2\lambda_4 r_1 + 2\lambda_6 r_3 \\
\frac{\partial \epsilon^2}{\partial r_3} &= \sum_{n=1}^{N} 2\Big(x_{n3} - \overline{x}_3 - r_3'(y_n - \overline{y})\Big)\,(y_n - \overline{y})\,(-1) \\
&\quad + 2\lambda_3 r_3 + 2\lambda_5 r_1 + 2\lambda_6 r_2
\end{aligned}$$

Setting these partial derivatives to zero and rearranging we obtain

$$\begin{aligned}
\sum_{n=1}^{N}(y_n - \overline{y}(y_n - \overline{y})'r_1 + \lambda_1 r_1 + \lambda_4 r_2 + \lambda_5 r_3 &= \sum_{n=1}^{N}(x_{n1} - \overline{x}_1)(y_n - \overline{y}) \\
\sum_{n=1}^{N}(y_n - \overline{y})(y_n - \overline{y})'r_2 + \lambda_4 r_1 + \lambda_2 r_2 + \lambda_6 r_3 &= \sum_{n=1}^{N}(x_{n2} - \overline{x}_2)(y_n - \overline{y}) \qquad (a) \\
\sum_{n=1}^{N}(y_n - \overline{y})(y_n - \overline{y})'r_3 + \lambda_5 r_1 + \lambda_6 r_2 + \lambda_3 r_3 &= \sum_{n=1}^{N}(x_{n3} - \overline{x}_3)(y_n - \overline{y})
\end{aligned}$$

Let $A = \sum_{n=1}^{N}(y_n - \overline{y})(y_n - \overline{y})'$, $\Lambda = \begin{pmatrix} \lambda_1 & \lambda_4 & \lambda_5 \\ \lambda_4 & \lambda_2 & \lambda_6 \\ \lambda_5 & \lambda_6 & \lambda_3 \end{pmatrix}$ and $B = (b_1 b_2 b_3)$. Then Eq.(a) can be simply rewritten as $A\,R' + R'\Lambda = B$. Multiplying both sides of Eq.() by R we have $R\,A\,R' + \Lambda = RB$. Since $A = A'$, $(RAR')' = RAR'$. Since both RAR' and Λ are symmetric, the left hand side must be symmetric. Hence, the right hand side is also symmetric. This means, $R\,B = (RB)'$ The solution for R now comes quickly. Let the singular value decomposition of B be $B = UDV$ where U and V are orthonormal and D is diagonal. Then $RUDV = (UDV)'R' = V'DU'R'$. By observation, a solution for R is immediately obtained as $R = V'U'$.

3.3 Experimental Results

Over 144,000 simulation experiments were done in which 3D points were chosen at random. A random rotation and translation are chosen and a corresponding point data set was created by rotating and translating the initial set of points and adding noise. The rotation and translation was then estimated.

The number of corresponding point pairs was varied between 10 and 200 in 9 steps. The signal to noise ratio, which is defined as 20 log (range of 3D points/standard deviation of noise), was varied between 8 db and 70 db in 8 steps. The noise distribution type was varied between Gaussian and Uniform. For each calculation one thousand trials were run.

Figure 2 illustrates a typical experimental result. It shows the mean angle error of the rotation, in degrees, as a function of signal to noise ratio with Gaussian noise. The plot indicates that when the number of 3D points is 25, then the RMS error of the rotation will be less than 3 degrees when the signal to noise ratio is greater than 25 db.

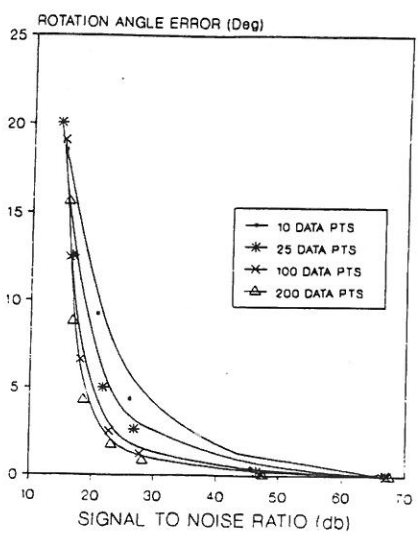

Figure 2. *Mean rotation angle error versus signal to noise ratio with Gaussian noise. Corresponding point data set sizes vary between 10 and 200 pairs. Each point on the graph represents 1,000 trials.*

Figure 3 shows rotation angle error plotted as a function of number of points in the coresponding point data sets for varying levels of Gaussian noise. This plot clearly shows that when the number of corresponding point data pairs is below 40, the estimated values are unreliable. When the number of corrsponding point data pairs is above 40, the estimates improve for increasing- sized sets.

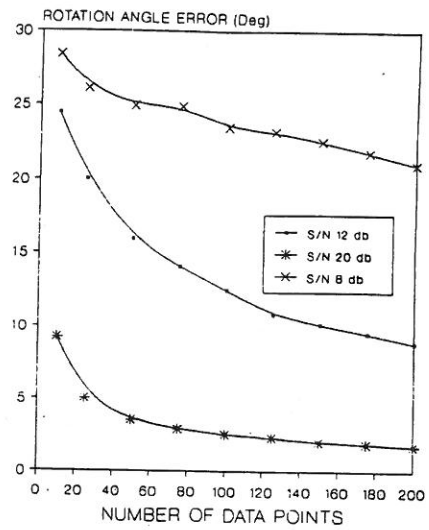

Figure 3. *Mean rotation angle error versus number of points with Gaussian noise.*

4. 2D Perspective–2D Perspective Projection Pose Estimation

The estimation of three-dimensional motion parameters of a rigid body is an important problem in motion ayalysis (Tsai and Huang[1]). Its applications include scene analysis, motion prediction, robotic vision, and dynamic industrial process.

There has been much literature contributing to 3D parameter estimation, but few of these contributions systematically discuss the effect of noise. Roach and Aggarwal[3] develop a nonlinear algorithm and deal with noisy data. Their results show that accuracy is improved by increasing the number of correspondence point pairs; but the number of corresponding point paris in their experiments is too few (15 corresponding point pairs). J.Q. Fang and T.S. Huang[4] use real world images to measure the noise effect on 3D parameter estimation. However, in order to get good matching, only a limited range of rotation angles is used, and it also has some constraints in translation range. In this paper we relax the rotation angle range and translation constraint. The relation between the error and noise level is obtained under a variety of corresponding point pairs. A simplified linear algorithm presented by Zhuang and Haralick[2] is used to get 3D motion parameters.

4.1 Simplified Linear Algorithm

In this section we review a simplified algorithm to determine the 3D motion parameters of a rigid body from two sequential perspective views. The rigid body is in motion in the $z < 0$ half plane. Let the two views be taken at t_1 and t_2 respectively, $t_1 < t_2$. Therefore we have $P_1 = (x_1, y_1, z_1)' =$ object space coordinate of the point at time t_1, $P_2 = (x_2, y_2, z_2)' =$ object space coordinate of the point at time t_2, $(X_1, Y_1) =$ image point at t_1, and $(X_2, Y_2) =$ corresponding image point at t_2. where $X_1 = x_1/z_1$, $Y_1 = y_1/z_1$, $X_2 = x_2/z_2$, $Y_2 = y_2/z_2$

The rigid body motion equation relating P_1 and P_2 is as follows:

$$P_2 = R_0 P_1 + T_0 \qquad (b)$$

where

$$R_0 = \begin{pmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{pmatrix}$$

$$T_0 = (t_{01} t_{02} t_{03})'$$

and also $R_0 R_0^t = I$ (I is a 3×3 identity matrix), T_0 is a 3×1 translation vector. If we use a camera 3D coordinate system centered at the optical center, then the Euler angles can be determined from R_0 as follows:

$$\begin{aligned}
r_1 &= \cos\psi \cos\theta \\
r_2 &= \sin\psi \cos\theta \\
r_3 &= \sin\theta \\
r_4 &= \sin\psi \cos\phi + \cos\psi \sin\phi \sin\phi \\
r_5 &= \cos\psi \cos\phi + \sin\psi \sin\theta \sin\phi \\
r_6 &= \cos\theta \sin\phi \\
r_7 &= \sin\psi \sin\phi + \cos\psi \sin\theta \cos\phi \\
r_8 &= -\cos\psi \sin\phi + \sin\psi \sin\theta \cos\phi \\
r_9 &= \cos\theta \cos\phi
\end{aligned}$$

4.2 The Two View Motion Equation

Recall that the 3D coordinates of a point before and after motion are related by Eq.(b). Taking any nonzero vector T which is colinear with T_0 and taking its cross-product with both sides of Eq.(b), we obtain $(z_2/z_1)T \times (X_2, Y_2, 1) = T \times [R_0(X_1, Y_1, 1)]$ and after taking the inner product of both sides of this equation with $(X_2, Y_2, 1)$, $(X_2, Y_2, 1)(T \times R_0)(X_1, Y_1, 1)' = 0$ Define the motion parameter matrix E by $E = T \times R_0$ This equation states that for any image corresponding point pair (X_1, Y_1), (X_2, Y_2) the 3×3 matrix satisfies the following Two–View Motion Equation which is linear and homogeneous in the nine elements of E:

$$(X_2, Y_2, 1) E (X_1, Y_1, 1) = 0 \qquad (c)$$

Any $T \times R_0$ with $T \times T_0 = 0$ satisfies Eq.(c). Moreover, such a colinear vector T has one degree of freedom when $T_0 \neq 0$ or three degrees of freedom when $T_0 = 0$. Thus the general solution of the Two–View Motion Eq.(c) has at least one degree of freedom when $T_0 \neq 0$ or three degrees of freedom when $T_0 = 0$.

As mentioned above, $T_0 \neq 0$ must have a rank 8, and $T_0 = 0$ must have rank 6. Under the surface assumption (Zhuang, Haralick, and Huang[2]) the number of image corresponding point pairs must be 8 when $T_0 \neq 0$, or greater than or equal to 6 when $T_0 = 0$. The geometry interpretation we use assumes that the object is stationary and the camera is moving. Let the origin of the camera system be O and O' respectively before and after motion. Then the surface assumption holds if and only if the 3D points corresponding to the observed image points do not lie on a quadratic surface passing through O and O' when $T_0 \neq 0$ or a cone with its apex at O when $T_0 = 0$.

4.3 Decomposing E

E has two decompositions; $T \times R_0(-T) \times R_0'$ with R_0 being an orthonormal matrix of the first kind. In order to determine the correct decomposition we note that $E = [T_{xr1}, T_{xr2}, T_{xr3}$. Hence, its three columns span a 2D space and also $\|E\| = \sqrt{2}\|T\|$. Therefore we can get three constraints as follows: Rank $(E) = 2$, $\|E\| = 2\|T\|$, $E'T = 0$ We can use the least square method to solve this equation for T and obtain the value of the T vector from the other two constraints. Since T is colinear with T_0, T_0 should have the same orientation as T or $-T$. Taking a cross-product with both sides of Eq.(b) by $(X_2 Y_2, 1)$ we obtain $z_1(X_2, Y_2, 1)' \times [R_0(X_1, Y_1, 1)] + (X_2, Y_2, 1) \times T_0 = 0$ Since $z < 0$, it implies that T_0 has the same orientation as T or $(-T)$ if and only if $(X_2, Y_2, 1)' \times [R_0(X_1, Y_1, 1)']$ has the same orientation as $(X_2, Y_2, 1) \times T$ or $[(X_2, Y_2, 1)' \times T]$. This implies it has the same orientation if and only if $\sum_{i=1}^{n} \left(\left\{ (X_2, Y_2, 1)' \times [R_0(X_1, Y_1, 1)'] \right\}' (X_2, Y_2, 1)' \times T > \right) \geq 0$ or ≤ 0. Once the correct T is determined, the true R_0 could be uniquely determined through $E = T \times R_0$ as follows: $R_0 = [E_2 \times E_3, E_3 \times E_1, E_1 \times E_2] - T \times E$ where $E = [E_1, E_2, E_3]$

4.4 Experimental Results

By mapping 3D spatial coordinates into image frame, and then adding noise to the points before and after motion, we obtain

$$\begin{pmatrix} X(t) \\ Y(t) \end{pmatrix} = \begin{pmatrix} 1/z(t) & 0 & 0 \\ 0 & 1/z(t) & 0 \end{pmatrix} \begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} + \begin{pmatrix} n_x(t) \\ n_y(t) \end{pmatrix}$$

Signal is related to object image size, and noise may come from camera error, digitization, or corresponding point extraction error.

In the simulation experiments, the 3D spatial coordinates before motion (x_1, y_1, z_1), true rotation matrix, and true translation vector are generated by a random number generator. Then the 3D spatial coordinates after motion (x_2, y_2, z_2) are calculated in the natural way. Projecting the 3D spatial coordinates into the image frame we get

image coordinates. Noisy image data is obtained by adding Gaussian or Uniform noise with zero mean to the image coordinates. The rotation matrix and the translation vector are then determined by the simplified linear algorithm. To make sure the corresponding point pairs follow the surface assumption, the generated data are checked before being used. As the number of corresponding point pairs increases the probability of the set not satisfying the surface assumption is very small.

In the first set of 64,000 experiments, the object image size is a square whose sides extend from -2 to 2. The signal to noise ratio (SNR) changes from 92 db to 26 db in 8 steps corresponding to a noise standard deviation varying from 0.0001 to 0.2. The Euler angles vary from -15 to 15 degrees. The number of corresponding point pairs varies from the 8-point pairs to 110-point pairs in 4 steps. One thousand trials are done for each variation. The standard deviation of the trial results are negligible for SNR greater than 70 db.

The results are shown in Figure 4. When noise-free, the error between the calculated value and true value is zero for all cases. Depending on kind of noise and number of corresponding point pairs, the error increases very rapidly when the signal to noise ratio gets below a knee value. Table 1 shows the minimum signal to noise ratio to guarantee a less than 1 degree error on a function of numbers of corresponding point pairs and kind of noise distribution. Other experiments with larger data sets show that the error is reduced by increasing the number of corresponding point pairs. However, the accuracy does not increase much as the number of corresponding point pairs increases from 200 to 500.

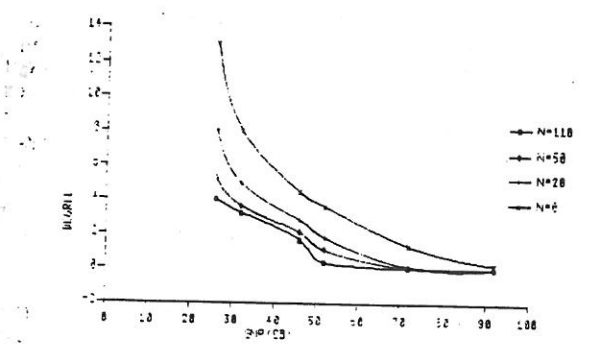

Figure 4. *Mean angle error between the estimated rotation angles and the true rotation angles versus the Gaussian noise level for four corresponding point data set sizes of 4 to 110 pairs. Each point on the graph represents 1,000 trials.*

	Rotation Angles			
No. of Point Pairs	8	20	50	110
Gausian	75	57	52	50
Uniform	74	56	52	49

	Translation Vector			
No. of Point Pairs	8	20	50	110
Gausian	105	78	73	68
Uniform	106	78	72	68

Table 1. *SNR (db) for error mean in 1 degree.*

5. Conclusion

We have presented solutions to three of the four pose estimation problems and have characterized the performance of these algorithms in simulation experiments with the noise model being additive Gaussian noise or Uniform noise. We have observed in these experiments a knee phenomenon. When the signal to noise ratio gets to be below a knee, the RMS error skyrockets. When the number of corresponding point pairs gets to be below a knee value, the RMS error also skyrockets.

Future simulation will be involved in broadening the noise model to include slash noise, Cauchy noise, and noise due to a fraction of the corresponding point pairs being incorrectly matched. Future statistical work will be in making the least squares solutions robust. This will involve using an iterative weighted least square technique where the weights at each iteration are related to the residual errors.

References

1. Tsai, R.Y., and T.S. Huang, "Uniqueness and estimation of 3D motion parameters of rigid objects with curved surfaces", *IEEE Trans. Pattern Analy. Mach. Intell.*, PAMI-6, 1984, pp. 13-17.
2. Zhuang, X., Haralick, R., and T.S. Huang, "Two-view motion analysis: a Unified Algorithm", *Opt. Soc. Am.*, Vol.3, No.9, 1986, pp. 1492-1450.
3. Roach, J.W., and J.K. Aggarwal, "Determining the movement of objects from a sequence of images", *IEEE Trans. Pattern Analy. Mach. Intell.*, PAMI-6, 1980, pp. 554-562.
4. Fang, J.Q., and T.S. Huang, "Some experimentson estimating the 3D motion parameters of a rigid body from two consecutive image frames", *IEEE Trans. Pattern Analy. Mach. Intell.*, PAMI-6, 1984, pp. 547-554.
5. Zhuang, X. and R. Haralick, "A Simplification Linear Algorithm", to be published.