

NASA-CR-196930

1N-63-CK
OCIT

NASA-629

24947

24P

SELF CALIBRATING AUTOTRAC A FINAL REPORT

Prepared for: Leo Monford, NASA-JSC

by
L. J. Everett
Texas A&M University

September 21, 1994

N95-11998

Unclass

G3/63 0024947

(NASA-CR-196930) SELF CALIBRATING
AUTOTRAC Final Report (Texas A&M
Univ.) 24 p

CONTENTS

ABSTRACT	1
INTRODUCTION	1
FORMULATING THE PROBLEM	2
THE GENERAL SOLUTION	4
Solving for the Rotation Part of X , R_x	5
Solving for the Translation Part of X	8
Solving for the Translation When Rotation Magnitude is Unknown	10
ALGORITHM	12
To Compute Rotation	12
To Compute Translation - Method 1	13
To Compute Translation - Method 2	14
EXAMPLE	14
Method 2 - When Robot Rotations are Unknown	15
Computing the Unit Vectors of R_x	17
Method 1 - When Robot Rotations Are Known	18
CONCLUSIONS	20
REFERENCES	21

ABSTRACT

The work reported here demonstrates how to automatically compute the position and attitude of a TRAC camera relative to the robot end effector. In the robotics literature this is known as the sensor registration problem. The registration problem is important to solve if TRAC images need to be related to robot position. Previously, when TRAC operated on the end of a robot arm, the camera had to be precisely located at the correct orientation and position. If this location is in error, then the robot may not be able to grapple an object even though the TRAC sensor indicates it should. In addition, if the camera is significantly far from the alignment it is expected to be at, TRAC may give incorrect feedback for the control of the robot. A simple example is if the robot operator thinks the camera is right side up but the camera is actually upside down, the camera feedback will tell the operator to move in an incorrect direction.

The automatic calibration algorithm requires the operator to translate and rotate the robot arbitrary amounts along (about) two coordinate directions. After the motion, the algorithm determines the transformation matrix from the robot end effector to the camera image plane.

This report discusses the TRAC sensor registration problem. Different aspects of this work has been published in [1, 2, 3] and submitted for publication in [4].

INTRODUCTION

When you mount a sensor on a robot it becomes necessary to find the pose (orientation and position) of the sensor relative to the robot. This is the sensor registration problem.

Many researchers have provided closed-form solutions to the sensor registration

problem; however the published solutions apply only to: (1) sensors that can measure a complete pose (three positions and three orientations) and (2) robots that can move precise distances. Neither of these requirements exists for our application.

Unless the TRAC is aligned with a mirror, it can provide only position information. If you assume the registration problem has not yet been solved, then it may not be possible to move the robot to align to a mirror.

If the robot is actually a velocity commanded telemanipulator (as in the SRMS), it is possible to enter a joystick command to move along an axis, but the precise distance moved may be awkward to determine. As a result it is necessary to solve the registration without precise knowledge of the robot's motion.

This report provides a closed-form solution to the sensor registration problem applicable when: (1) the sensor can provide at least position information, and (2) the robot can move on and rotate about straight lines.

FORMULATING THE PROBLEM

This report assumes ${}^uT^v$ is a 4 by 4 homogeneous transformation matrix describing the coordinate transformation from frame u to frame v , this is equivalent to the pose of v relative to the pose of u . Also, this report assumes sensors, such as a range sensor, a camera, or a fixture, are fixed relative to tool, t , while in use but may be removable. Thus, the frame of the sensor relative to the tool (represented as ${}^tT^s$ which is what we desire to determine) is constant while in use.

We will discuss the pose of various objects at several manipulator positions. To designate what robot position we are referring to, we will use a subscript on the transformation matrix. Some matrices are constant regardless of the robot position. Constant transformations will have no subscripts. For example, to designate a trans-

formation (${}^uT^v$) when the robot is at location i , we will use ${}^uT_i^v$. The transform when the robot is at location j is ${}^uT_j^v$. The relative change in the transform relative to u is:

$${}^uT_i^v ({}^uT_j^v)^{-1} = {}^uT_{v_i \rightarrow j}^u$$

In an application, suppose a robot has a feedback sensor mounted on its tool. When the robot is positioned at location i , the pose relative to the sensor frame of an object fixed in g is represented with the matrix ${}^gT_i^o$. Furthermore, we presume that the tool relative to the sensor frame, (${}^gT^t$), is unknown but desired. The problem is formulated as follows:

$${}^gT^o = {}^gT_i^{tt} T^{ss} T_i^o \quad (1)$$

Now, if the manipulator is positioned at j , we have the following:

$${}^gT^o = {}^gT_j^{tt} T^{ss} T_j^o$$

As shown in the transform graph of Figure 1, we can formulate the following:

$${}^i T_i^{gg} {}^g T_j^{tt} T^{ss} = {}^i T^{ss} T_i^{oo} T_j^s \quad (2)$$

Equation 2 is of the form $AX = XB$ which shows it to be the sensor registration problem [6] and [7, 8] also [9, 10]. The B transform is typically measured, while the A transform is computed from joint angles. The objective is to solve for X given A and B . As shown in Figure 1, the transformations A and B describe the relative displacements of an arbitrary movement (from i to j) of the tool frame (${}^gT^t$) and sensor frame (${}^gT^o$), respectively. The constant matrix X in Figure 1, is the relative transformation between the tool frame and the sensor frame. The published solutions for X , including the analytical ones, are valid only if: (1) the sensor provides complete pose¹ data and the robot provides complete tool pose accurately. This full

¹Complete pose measurement requires three position and three orientation components.

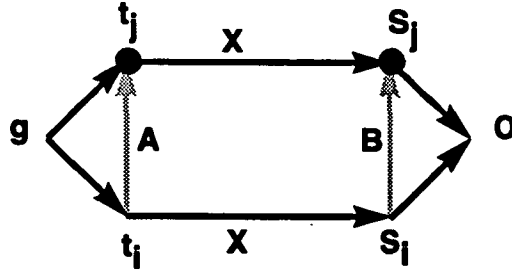


Figure 1: An Arbitrary Movement of the Tool of a Robot and Formulating a Homogeneous Transform Equation of the Form $AX = XB$.

pose requirement means that matrices A and B are completely known and invertible. Thus, in situations, where either A or B is not fully known, the registration problem is more complex, and the published solutions cannot be applied.

THE GENERAL SOLUTION

To find a solution to the registration problem, we need to manipulate the parts of the A , B and X matrices given in equation 2. The parts of the A matrix will be referenced as follows:

$$A = \begin{bmatrix} R_a & P_a \\ 000 & 1 \end{bmatrix} = \begin{bmatrix} {}^tR_{g_i \rightarrow j}^t & {}^tP_{g_i \rightarrow j}^t \\ 000 & 1 \end{bmatrix} \quad (3)$$

where ${}^tP_{g_i \rightarrow j}^t$ is the 3 by 1, translation vector and ${}^tR_{g_i \rightarrow j}^t$ represents a 3 by 3 rotation part of A . The rotation and translation components of B will be represented as ${}^sR_{o_i \rightarrow j}^s$ and ${}^sP_{o_i \rightarrow j}^s$; the parts of ${}^uT_i^v$ are ${}^uR_i^v$ and ${}^uP_i^v$. The parts of X will be expressed as:

$$X = \begin{bmatrix} R_x & P_x \\ 000 & 1 \end{bmatrix}$$

We will need parts of ${}^uR^v$ so we will write:²

$${}^uR^v = \begin{bmatrix} {}^uN^v & {}^uO^v & {}^uG^v \end{bmatrix} = \begin{bmatrix} {}^uN^{vT} \\ {}^uO^{vT} \\ {}^uG^{vT} \end{bmatrix}^T \quad (4)$$

We can express the homogeneous transformation equation $AX = XB$ in terms of rotation and position transformation matrices as follows:

$$\begin{bmatrix} R_a & P_a \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_x & P_x \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_x & P_x \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_b & P_b \\ 0 & 1 \end{bmatrix} \quad (5)$$

Expanding equation 5, we have,

$$R_a R_x = R_x R_b \quad (6)$$

and,

$$(R_a - I_3) P_x = R_x P_b - P_a \quad (7)$$

where I_3 is a 3 by 3 identity matrix.

Solving for the Rotation Part of X , R_x

Lemma 1 *If the robot translates the tool frame t , from location i , an amount (x, y, z) relative to frame t , then:*

$$A = \begin{bmatrix} I_3 & \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ 000 & 1 \end{bmatrix}$$

²It is common practice to express the rotational components as N (normal), O (orthogonal), and A (approach) directions. Using A for the third direction however, might cause confusion with the A matrix, the common notation used in the sensor registration problem. As a result, we must adopt a different notation for one or the other. We chose to call the third rotational vector, the Grip direction G .

First, suppose the robot is located at location i given by: ${}^sT_i^t$. Next allow the robot to move to location j by translating (x, y, z) ; therefore, we can write:

$${}^sT_j^t = {}^sT_i^t \begin{bmatrix} I_3 & \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ 000 & 1 \end{bmatrix}$$

This can be written as:

$${}^tT_i^s {}^sT_j^t = \begin{bmatrix} I_3 & \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ 000 & 1 \end{bmatrix}$$

By the definition of A , we have:

$$A = \begin{bmatrix} I_3 & \begin{bmatrix} x \\ y \\ z \end{bmatrix} \\ 000 & 1 \end{bmatrix} \text{ Q.E.D.}$$

Lemma 2 *If the robot translates the tool frame t , from location i , an amount (x, y, z) relative to frame t , then:*

$$B = \begin{bmatrix} I_3 & {}^sP_i^o - {}^sP_j^o \\ 000 & 1 \end{bmatrix}$$

Let the tool be located at i and the object's location relative to the sensor be:

$${}^sT_i^o = \begin{bmatrix} {}^sR_i^o & {}^sP_i^o \\ 000 & 1 \end{bmatrix}$$

Suppose the tool frame is translated until it reaches location j and the object relative to the sensor is given by:

$${}^sT_j^o = \begin{bmatrix} {}^sR_j^o & {}^sP_j^o \\ 000 & 1 \end{bmatrix}$$

By definition we have $B = {}^sT_i^{oo}T_j^s$, from lemma 1 we know ${}^tR_{g_i \rightarrow j}^t = I_3$, hence from equation 6 we can write:

$$R_x = R_x {}^sR_{o_i \rightarrow j}^s$$

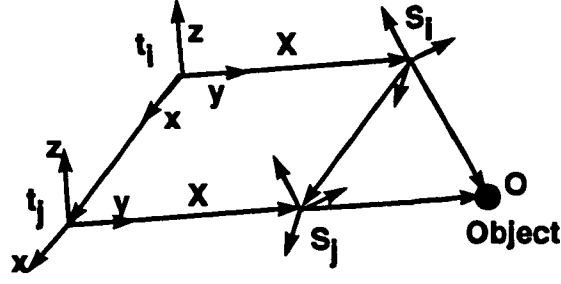


Figure 2: The Tool Translates Along Its x-axis; to compute the Unit Vector N_x^T of ${}^sT^t$.

Premultiplying by the inverse of R_x , we find $I_3 = {}^sR_{o_i \rightarrow j}^s$. From the definition of B , we can expand the multiplication to express the translational part of B as ${}^sP_{o_i \rightarrow j}^s = {}^sR_i^{oo} P_j^s + {}^sP_i^o$. Now from the definition of the inverse of a transformation matrix we can write:

$${}^sT^o = ({}^oT^s)^{-1} = \begin{bmatrix} {}^sR^o & -{}^sR^{oo}P^s \\ 000 & 1 \end{bmatrix} \quad (8)$$

Using the last two expressions together, we have ${}^sP_{o_i \rightarrow j}^s = -{}^sP_j^o + {}^sP_i^o$. Q.E.D.

Now we will use these lemmas to solve for the rotational parts of X . As shown in Figure 2 equation 2, and lemma 1, we observe that if we translate an unknown distance Δ_x along the x-axis of t , we can write equation 7 as:

$$0 = R_x {}^sP_{o_i \rightarrow j}^s - \begin{bmatrix} \Delta_x \\ 0 \\ 0 \end{bmatrix}$$

If we express R_x with its component parts, we have:

$$\begin{bmatrix} \Delta_x \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} N_x \cdot {}^sP_{o_i \rightarrow j}^s \\ O_x \cdot {}^sP_{o_i \rightarrow j}^s \\ G_x \cdot {}^sP_{o_i \rightarrow j}^s \end{bmatrix} \quad (9)$$

where ${}^sP_{o_i \rightarrow j}^s = {}^sP_i^o - {}^sP_j^o$ is the relative position vector of the sensor frame from position i to position j and is measurable. Equation 9 says that ${}^sP_{o_i \rightarrow j}^s$ is perpendicular to O_x and G_x . Therefore, ${}^sP_{o_i \rightarrow j}^s$ (which is known from measurements) must be in the direction of N_x . This implies N_x equals a unit vector along ${}^sP_{o_i \rightarrow j}^s$.

Similarly, we can compute O_x by translating the robot tool frame along the y axis of the tool an arbitrary unknown amount, Δ_y . The third orientation quantity, G_x , can be computed easily by translating the tool frame along the z-axis, or by using the orthogonal property of R_x , that is, $G_x = N_x \times O_x$.

Solving for the Translation Part of X

Lemma 3 *If the robot rotates the tool frame t , from location i , an amount given by R , about an axis of t , then:*

$$A = \begin{bmatrix} R & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ 000 & 1 \end{bmatrix}$$

First, suppose the robot is located at location i given by: ${}^gT_i^t$. Next allow the robot to move to location j by rotating about an axis through the origin of t . We can write:

$${}^gT_j^t = {}^gT_i^t \begin{bmatrix} R & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ 000 & 1 \end{bmatrix}$$

This can be written as:

$${}^tT_i^g {}^gT_j^t = \begin{bmatrix} R & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \\ 000 & 1 \end{bmatrix} = A$$

Q.E.D.

Lemma 4 *If the robot rotates the tool frame t , from location i , an amount given by R , about an axis of t , then:*

$$B = \begin{bmatrix} \{(R_x)^{-1} R R_x\} & \{ {}^sP_i^o - (R_x)^{-1} R R_x {}^sP_j^o \} \\ 000 & 1 \end{bmatrix}$$

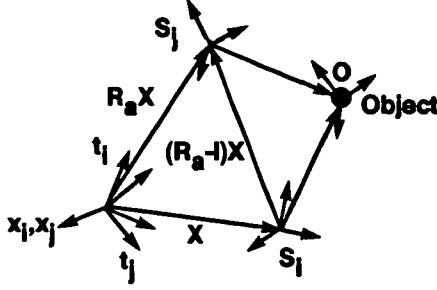


Figure 3: The Tool Rotates About Its x Axis; to Compute the y and z Components of P_x .

From equation 6, we have ${}^s R_{o_i \rightarrow j}^s = (R_x)^{-1} {}^t R_{g_i \rightarrow j}^t R_x$. Using lemma 3 we find ${}^s R_{o_i \rightarrow j}^s = (R_x)^{-1} R R_x$. From the definition of B , we have:

$${}^s P_{o_i \rightarrow j}^s = {}^s R_i^{oo} P_j^s + {}^s P_i^o = {}^s R_i^{oo} R_j^{ss} R_j^{oo} P_j^s + {}^s P_i^o = {}^s R_{o_i \rightarrow j}^s ({}^s R_j^{oo} P_j^s) + {}^s P_i^o$$

Using equation 8 we can write:

$${}^s P_{o_i \rightarrow j}^s = -{}^s R_{o_i \rightarrow j}^s {}^s P_j^o + {}^s P_i^o$$

Q.E.D.

Now we will determine P_x . When ${}^t R_{g_i \rightarrow j}^t$ is not equal to I_3 , it has one and only one eigenvalue equal to 1; therefore, $({}^t R_{g_i \rightarrow j}^t - I_3)$ in equation 7 has a rank equal to two [6]. Therefore, P_x in equation 7 has one degree of freedom, and we can use it to solve for two components of P_x .

For example, if we rotate the tool frame with an amount θ_{tx} from location i about the x -axis of t , then ${}^t P_{g_i \rightarrow j}^t = 0$ and equation 7 appears as follows (by lemma 3):

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & \cos(\theta_{tx}) - 1 & -\sin(\theta_{tx}) \\ 0 & \sin(\theta_{tx}) & \cos(\theta_{tx}) - 1 \end{bmatrix} \begin{bmatrix} P_x \end{bmatrix} = R_x {}^s P_{o_i \rightarrow j}^s \quad (10)$$

Figure 3 shows the transformation graph described by equation 10. Equation 10 has no information for finding the x component of P_x . However, we can resolve the y

and z components by equating the last two non-zero rows of equation 10. Note that provided the angle θ_{tx} is known, everything in equation 10 except P_x is known. For example, R_x has been determined already, and ${}^sP_{o_i \rightarrow j}$ comes from lemmas 3 and 4.

Equations 7 and 10 show that $(R_a - I_3)$ becomes zero if the magnitude of rotation about any axis is zero or a multiple of 2π . Therefore, this condition should be avoided when solving the translation part of ${}^tT^s$.

Similarly, you can compute the x component of P_x by rotating the tool frame about either the y or z axis of frame t , then using the two non-zero rows of the equation resulting from 7.

Solving for the Translation When Rotation Magnitude is Unknown

During the investigation, we discovered another, more general, method for finding the translational components when the robot rotations are unknown. The translational components can be determined after performing a minimum of three rotations about the tool frame.

The left of Figure 4 shows that a rotation about the tool frame changes only the orientations of the tool and sensor frames, and not the object frames. This is due to the fact that the object is fixed. The right half of Figure 4 shows the effect of the same rotation relative to the tool frame. The position of the object relative to the sensor in both figures are the same. For example, let ${}^gT_0^t$ represent the position of the tool frame (relative to the ground) before any rotations are performed, and ${}^gT_i^t$ is after rotation i . Now suppose the robot is moved so it rotates an angle θ about the n axis of the tool (neither θ nor n need to be known). We will represent the motion with the transform $R_{\theta,n}$. After application of the motion, the object's origin can be

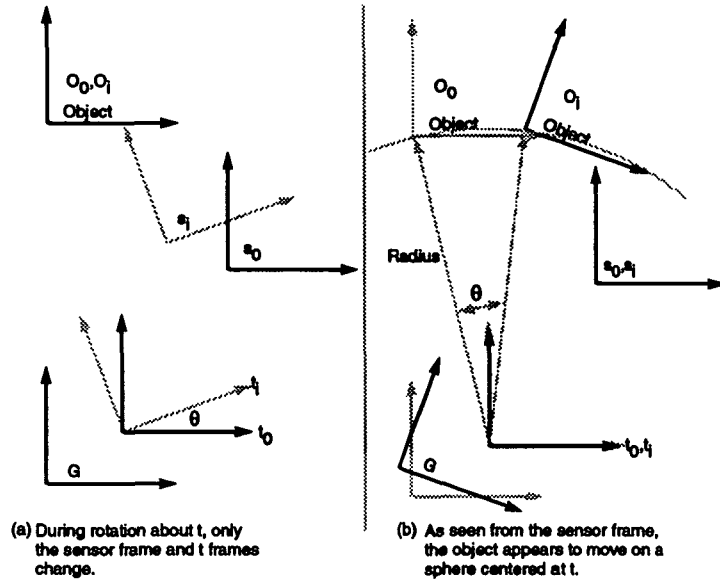


Figure 4: The Effect of a Rotation About the Tool Frame.

computed as:

$${}^tP_i^o = R_{-\theta, n} {}^tP_0^o$$

which demonstrates ${}^tP_i^o$ to be a point on a sphere centered at the origin of the tool. If four points on this sphere can be measured (implying a minimum of three rotations), the sphere's center (the origin of the tool) can be found.

Since the sensor measures the position of the object relative to the sensor frame, measuring the object before and after the tool rotations enables the tool center relative to the sensor frame ${}^sP^t$ to be computed. Once the center is found, ${}^tP^s$ is computed as:

$${}^tP^s = -{}^tR^{ss}P^t \quad (11)$$

Computing the center of a sphere given four points is a simple and well defined problem. Singularities in the problem can be avoided if all four points are distinct and do not lie on a single plane. The method we used to compute the sphere center is performed using the following steps.

1. Number the points 1, 2, 3, and 4. A point is a 1 by 4 matrix, the first 3 elements are the x, y, z coordinates, the fourth element is 1. Let P_i represent point i .
2. Pair the points into three groups such as (1,2), (2,3), (3,4).
3. Compute midpoints (m_i) of each point group:

$$m_1 = \frac{P_1 + P_2}{2}$$

4. Compute unit vectors (v_i) between the points in each pair:

$$v_1 = \frac{P_2 - P_1}{|P_2 - P_1|}$$

5. Compute the equation of planes (C_i) normal to v_i containing m_i :

$$C_1 = (v_1, 0) - (0, 0, 0, v_1 \cdot m_1)$$

6. Solve the linear equations for the sphere center (x, y, z):

$$\begin{bmatrix} C_1 \\ C_2 \\ C_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

ALGORITHM

This section summarizes the discussion in the previous sections and puts the material into algorithmic steps.

To Compute Rotation

To compute the rotation component of matrix ${}^sT^t$, translate the tool frame along two different axes, one at a time.

1. Record the position of the object with respect to the sensor at position 1, ${}^sP_1^o$.

2. Translate the tool in the positive x-axis to position 2. Record the new position of the object with respect to the sensor, ${}^sP_2^o$.
3. Translate the tool in the positive y-axis to position 3 and record the position, ${}^sP_3^o$.
4. Compute the unit vectors of the rotation matrix of the sensor-to-tool frame, ${}^sR^t$, using:

$${}^sN^t = \frac{{}^sP_1^o - {}^sP_2^o}{|{}^sP_1^o - {}^sP_2^o|} \quad (12)$$

$${}^sO^t = \frac{{}^sP_2^o - {}^sP_3^o}{|{}^sP_2^o - {}^sP_3^o|} \quad (13)$$

and

$${}^sG^t = {}^sN^t \times {}^sO^t \quad (14)$$

5. Compute the rotation matrix of the tool-to-sensor frame: ${}^tR^s = ({}^sR^t)^T$.

To Compute Translation - Method 1

To compute the translation part of ${}^sT^t$, rotate the tool frame about any two of its axes, one at a time.

1. Compute the global to tool frame at position 4, (this is done using a forward kinematic model and joint location measurements) ${}^gT_4^t$, and record ${}^sP_4^o$.
2. Rotate the tool about its x-axis to position 5. Compute ${}^gT_5^t$ and record ${}^sP_5^o$.
3. Compute the relative rotation of the tool frame from position 4 to 5, ${}^tR_{g4 \rightarrow 5}^t = \text{rotation part of } ({}^gT_4^t)^{-1} {}^gT_5^t$.
4. Compute the relative rotation of the sensor frame, ${}^sR_{o4 \rightarrow 5}^s = R_x^{-1} {}^tR_{g4 \rightarrow 5}^t R_x$.

5. Compute:

$${}^s P_{o_4 \rightarrow s}^s = -{}^s R_{o_4 \rightarrow s}^s {}^s P_5^o + {}^s P_4^o \quad (15)$$

6. Compute the y and z components of P_x by using equation 10.

7. Rotate the tool frame about its y -axis to location 6 and repeat steps 2 through 5.

Use a variant of equation 10 to compute the x component of P_x .

To Compute Translation - Method 2

1. Record ${}^s_4 P^0$ at location 4.
2. Rotate about the x -axis of the tool to location 5 and record the objects' position, ${}^s_5 P^0$.
3. Rotate about the y -axis of the tool to location 6 and record the object's position, ${}^s_6 P^0$.
4. Rotate about the z -axis of the tool to location 7 and record the object's position, ${}^s_7 P^0$.
5. Compute ${}^s P^t$ at the sphere center of the above 4 points.
6. Compute ${}^t P^s = -{}^t R^{ss} P^t$.

EXAMPLE

We used a software simulation to test the algorithm proposed in this report. We imagine a position sensor on the last link of a robot manipulator for feedback control

application. The real position and orientation of the sensor frame relative to the tool frame (matrix X) is *chosen* as follows:

$${}^tT^s = \begin{bmatrix} 0.925 & -0.163 & 0.342 & 20 \\ 0.319 & 0.823 & -0.470 & -12 \\ -0.205 & 0.544 & 0.814 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The solution for X requires movements of the tool frame ${}^gT^t$: translations and rotations. In practice, the tool pose ${}^gT^t$ and the object's position relative to the sensor (${}^sP^o$ in this example) are determined from the information provided by the robot's controller and the sensor. In simulation, however, ${}^gT^t$ and ${}^sP^o$ are computed by using equations: ${}^gT_j^t = {}^gT_i^t {}^tT_{g_i \rightarrow j}^t$ and ${}^sP_j^o = ({}^gT_j^t {}^tT^s)^{-1} {}^sP^o$, where ${}^tT_{g_i \rightarrow j}^t$ is an *assumed* movement (either translation, rotation or both) of the tool frame from position i to position j , and ${}^sP^o$ is an assumed position relative to the global frame g of a fixed object o .

In the following simulation, the ${}^gT^o$ was chosen as:

$${}^gT^o = \begin{bmatrix} 1 & 0 & 0 & 20 \\ 0 & 1 & 0 & -15 \\ 0 & 0 & 1 & 10 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The initial robot position (${}^gT_1^t$) was chosen as:

$${}^gT_1^t = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & -5 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Method 2 - When Robot Rotations are Unknown

The required positions of the tool frame are chosen as: (1) Given above, (2) Translate 10 units along the x-axis from position 1, (3) Translate 5 units along the y-axis from position 2, (4) Equal to point 1, (5) Rotate 30 degrees about the x-axis

from position 4, and (6) Rotate 20 degrees about the y-axis from position 5, (7)

Rotate 40 degrees about the z-axis from position 6.

These positions are:

$${}^gT_1^t = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & -5 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^gT_2^t = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^gT_3^t = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & 8 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^gT_4^t = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & -5 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^gT_5^t = \begin{bmatrix} 0 & 0.5 & 0.866 & 2. \\ 1. & 0 & 0 & -5. \\ 0 & 0.866 & -0.5 & 3. \\ 0 & 0 & 0 & 1. \end{bmatrix}$$

$${}^gT_6^t = \begin{bmatrix} -0.296 & 0.5 & 0.814 & 2. \\ 0.940 & 0 & 0.342 & -5. \\ 0.171 & 0.866 & -0.470 & 3. \\ 0 & 0 & 0 & 1. \end{bmatrix}$$

$${}^gT_7^t = \begin{bmatrix} 0.0945 & 0.573 & 0.814 & 2. \\ 0.720 & -0.604 & 0.342 & -5. \\ 0.688 & 0.553 & -0.470 & 3. \\ 0 & 0 & 0 & 1. \end{bmatrix}$$

The sensor output corresponding to each of these positions are:

$${}^sP_1^o = \begin{bmatrix} -23.3 & 24.9 & -12.7 & 1 \end{bmatrix}$$

$${}^s P_2^o = \begin{bmatrix} -32.6 & 26.5 & -16.1 & 1 \end{bmatrix}$$

$${}^s P_3^o = \begin{bmatrix} -34.2 & 22.4 & -13.7 & 1 \end{bmatrix}$$

$${}^s P_4^o = \begin{bmatrix} -23.3 & 24.9 & -12.7 & 1 \end{bmatrix}$$

$${}^s P_5^o = \begin{bmatrix} -19.6 & 28.3 & -21.3 & 1 \end{bmatrix}$$

$${}^s P_6^o = \begin{bmatrix} -22.0 & 26.6 & -25.9 & 1 \end{bmatrix}$$

$${}^s P_7^o = \begin{bmatrix} -8.44 & 28.8 & -23.9 & 1 \end{bmatrix}$$

Computing the Unit Vectors of R_x

Using equations 12, 13 and 14, ${}^s N^t$, ${}^s O^t$, and ${}^s G^t$ are found using the translational components of positions 1, 2, and 3 as:

$${}^s N^t = \begin{bmatrix} 0.925 & -0.163 & 0.342 & 0 \end{bmatrix}^T$$

$${}^s O^t = \begin{bmatrix} 0.319 & 0.823 & -0.470 & 0 \end{bmatrix}^T$$

and

$${}^s G^t = \begin{bmatrix} -0.205 & 0.544 & -0.814 & 0 \end{bmatrix}^T$$

Using the orthonormal properties of R_x (eg. ${}^t R^s = {}^s R^{tT}$) we obtain:

$$R_x = \begin{bmatrix} 0.925 & -0.163 & 0.342 \\ 0.319 & 0.823 & -0.470 \\ -0.205 & 0.544 & 0.814 \end{bmatrix} \quad (16)$$

The translational components of points 4 through 7 lie on a sphere. The center of the sphere is computed to be:

$${}^s P^t = \begin{bmatrix} -12.6 & 7.70 & -20.62 & 1. \end{bmatrix}$$

Using equations 16 and 11, we determine ${}^t P^s$ as:

$${}^t P^s = -{}^t R^{ss} P^t = \begin{bmatrix} 20. & -12. & 10. & -1. \end{bmatrix}$$

Method 1 - When Robot Rotations Are Known

In the following simulation, the ${}^gT^o$ and initial positions ${}^gT_1^t$ are the same as before. The position ${}^gT_4^t$ (starting point for the rotations) was chosen to be equal to ${}^gT_1^t$ given above.

Translating the robot tool along the x-axis and y-axis, one at a time (same amounts as before), the respective positions, ${}^sP_2^o$ and ${}^sP_3^o$, are recorded as follows:

$${}^sP_1^o = \begin{bmatrix} -23.3 & 24.9 & -12.7 & 1 \end{bmatrix}^T$$

$${}^sP_2^o = \begin{bmatrix} -32.6 & 26.5 & -16.1 & 1 \end{bmatrix}^T$$

and

$${}^sP_3^o = \begin{bmatrix} -24.9 & 20.8 & -10.3 & 1 \end{bmatrix}^T$$

At an initial position 4, record ${}^gT_4^t$ and ${}^sP_4^o$. Rotate the tool about its x-axis a non-trivial amount and record ${}^gT_5^t$ and ${}^sP_5^o$. Rotate the tool from position 5 about its y-axis to position 6, and record ${}^gT_6^t$ and ${}^sP_6^o$. The resulting input values are as follows:

$${}^gT_4^t = \begin{bmatrix} 0 & 0 & 1 & 2 \\ 1 & 0 & 0 & -5 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^gT_5^t = \begin{bmatrix} 0 & 0.5 & 0.866 & 2 \\ 1 & 0 & 0 & -5 \\ 0 & 0.866 & -0.5 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^gT_6^t = \begin{bmatrix} -0.296 & 0.5 & 0.814 & 2 \\ 0.940 & 0 & 0.342 & -5 \\ 0.171 & 0.866 & -0.470 & 3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^sP_4^o = \begin{bmatrix} -24.3 & 24.4 & -24.9 & 1 \end{bmatrix}^T$$

$${}^s P_5^o = \begin{bmatrix} -22.1 & 22.5 & -31.8 & 1 \end{bmatrix}^T$$

and

$${}^s P_6^o = \begin{bmatrix} -20.5 & 19.6 & -35.8 & 1 \end{bmatrix}^T$$

Using equations 12, 13 and 14, ${}^s N^t$, ${}^s O^t$, and ${}^s G^t$ are found as:

$${}^s N^t = \begin{bmatrix} 0.925 & -0.163 & 0.342 & 0 \end{bmatrix}^T$$

$${}^s O^t = \begin{bmatrix} 0.319 & 0.823 & -0.470 & 0 \end{bmatrix}^T$$

and

$${}^s G^t = \begin{bmatrix} -0.205 & 0.544 & -0.814 & 0 \end{bmatrix}^T$$

Using the orthonormal properties of R_x (eg. ${}^t R^s = ({}^s R^t)^T$) we obtain:

$$R_x = \begin{bmatrix} 0.925 & -0.163 & 0.342 \\ 0.319 & 0.823 & -0.470 \\ -0.205 & 0.544 & 0.814 \end{bmatrix}$$

Compute the relative rotation matrices of the tool for the two rotations: that is the rotation part of ${}^t T_4^{gg} T_5^t$ and the rotation part of ${}^t T_5^{gg} T_6^t$. They are found as:

$${}^4 R^5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.866 & -0.5 \\ 0 & 0.5 & 0.866 \end{bmatrix}$$

and

$${}^5 R^6 = \begin{bmatrix} 0.940 & 0 & 0.342 \\ 0 & 1 & 0 \\ -0.342 & 0 & 0.940 \end{bmatrix}$$

By using equation 6 (eg. ${}^s R_{o_i \rightarrow j}^s = R_x^{-1t} R_{g_i \rightarrow j}^t R_x$), the relative rotations of the sensor are computed as:

$${}^s R_{o_4 \rightarrow s}^s = \begin{bmatrix} 0.981 & -0.191 & -0.0392 \\ 0.151 & 0.870 & -0.470 \\ 0.124 & 0.455 & 0.882 \end{bmatrix}$$

and

$${}^s R_{o_5 \rightarrow 6}^s = \begin{bmatrix} 0.946 & 0.176 & 0.272 \\ -0.145 & 0.980 & -0.132 \\ -0.290 & 0.0857 & 0.953 \end{bmatrix}$$

Using equations 15, compute ${}^s P_{o_4 \rightarrow 5}^s$ and ${}^s P_{o_5 \rightarrow 6}^s$, and their values are

$${}^s P_{o_4 \rightarrow 5}^s = \begin{bmatrix} 0.422 \\ -6.78 \\ -4.38 \end{bmatrix} \quad (17)$$

and

$${}^s P_{o_5 \rightarrow 6}^s = \begin{bmatrix} 3.57 \\ -4.41 \\ -5.30 \end{bmatrix} \quad (18)$$

Finally, compute the components of P_x by using equations 10, 17 and 18. We find $P_x = (20, -12, 10)^T$.

CONCLUSIONS

When a TRAC sensor is mounted on a robot for feedback control applications, the sensor measures objects relative to the sensor frame. Since the ultimate objective is to know the tool frame with respect to the object, it is necessary to compute the sensor frame relative to the tool. Determining the tool-to-sensor frame is a form of the sensor registration problem confronted by researchers using tool mounted feedback sensors. The published solutions to the sensor registration problem are applicable only when the sensor is able to measure a complete pose and the robot tool pose is fully known. However, unless the TRAC is nominally normal to the mirror target, it can only measure position. Additionally, in the case of velocity driven robots (like the SRMS), there may be no convenient access to precise tool pose information.

This report provides a closed-form solution for solving the sensor registration, ${}^t T^s$, when a position only sensor is used. The solution requires movements of the tool

frame ${}^gT^t$: translations and rotations.

The method has been demonstrated in simulation. Initial experiments have not been successful due to measurement errors. The work used a TRAC target at a 30 foot range. Initial simulation studies indicate the sensitivity will improve significantly with a shorter range. Future work will investigate the sensitivity of the method and results will be forwarded to the sponsor.

REFERENCES

- [1] L. J. Everett and L. Ong, "Closed form solution for the sensor registration problem using position information only," *Journal of Robotic Systems*, To Appear 1994.
- [2] L. J. Everett and L. Ong, "Closed form solutions for the sensor registration problem applicable to position sensors and velocity commanded robots," in *ASME Winter Annual Meeting, Book No. DSC-Vol 49*, pp. 287-294, November 1993.
- [3] L. J. Everett, "Calibrating a trac sensor," in *5th International Symposium on Robotics and Manufacturing - ISRAM'94, Volume 1, Intelligent Automation and Soft Computing*, pp. 63-68, August 1994.
- [4] L. J. Everett and L. Ong, "Closed form solutions for the sensor registration problem," *IEEE Transactions on Robotics and Automation*, Submitted in 1994.
- [5] R. P. Paul, *Robot Manipulators: Mathematics, Programming, and Control*. MIT Press, 1981.
- [6] Y. C. Shiu and S. Ahmad, "Finding the mounting position of a sensor by solving a homogeneous transform equation of the form $AX=Xb$," in *IEEE International Conference on Robotics and Automation, Vol. 3*, pp. 1666-1671, IEEE Computer Society, 1987.
- [7] J. C. K. Chou and M. Kamel, "Quaternions approach to solve the kinematic equation of rotation, $a_a a_z = a_z a_b$, of a sensor-mounted robotics manipulator," in *IEEE International Conference on Robotics and Automation, Vol. 2*, pp. 656-662, IEEE Computer Society, 1988.
- [8] J. C. K. Chou and M. Kamel, "Finding the position and orientation of a sensor on a robotics manipulator using quaternions," *The International Journal of Robotics Research*, vol. 10, pp. 240-254, June 1991.
- [9] C. C. Wang, "Extrinsic calibration of a vision sensor mounted on a robot," *IEEE Transactions on Robotics and Automation*, vol. 8, pp. 161-175, April 1992.

- [10] H. H. Chen, "Pose determination from line-to-plane correspondences: Existence condition and closed-form solutions," in *IEEE International Conference on Computer Vision*, (Osaka, Japan), pp. 374-378, 1990.