Robotica (1997) volume 15, pp 511–521. Printed in the United Kingdom © 1997 Cambridge University Press

Interactive building and updating of a 3D database for teleoperation

Edie Camille N'Zi, Malik Mallem and Florent Chavand

CEMIF Systèmes Complexes, 40 rue du Pelvoux, CE 1455 Courcouronnes, 91020 Evry Cedex (France)
Email: mallem@cemif.univ-evry.fr

(Received in Final Form: October 23, 1996)

SUMMARY

The paper deals with the modelling of unknown objects and location determination of known ones. The operator models objects using data issued from the video image and a time of flight infra red range finder. We present methods to build an embodying volume for unknown objects. These methods are available for polyhedral and conical objects, which represent the basis of most industrial environments. The location determination of known objects is determined using 2D clues obtained through video image. A two-step algorithm is implemented and assessed. The first step is available in case of object large motion, it is a geometric algorithm, which doesn't use redundant data; it provides a first estimate of the pose. The second step is available in case of object small motion. A linear approach is carried out to determine the pose. It allows the use of redundant data, so it improves the accuracy achieved after the first step.

KEYWORDS: Telerobotics; 3D databases; Images matching; Geometric algorithm.

I. INTRODUCTION

Teleoperated robots are remote-controlled, remote-monitored mechanical systems used for working in inaccessible or man-hostile environments, such as space, deep sea, areas contaminated by radioactivity or chemical products, military intervention zones, etc.[1]

Our laboratory is designing a multimedia control interface for teleoperation whose purpose is to provide the human operator with a visual aid for the perception of remote scenes and for robot control in case of indirect viewing via a video camera. Several teleoperated systems use geometrical 3D data:[2,3]

The devices consist of a camera which gives the human operator an indirect view of a scene (real world); proprioceptive and exteroceptive sensors allowing the building of the 3D geometric database of an environment (virtual world). The virtual world is projected onto a video display terminal. Computer-generated and video images are superimposed. The man-machine interface functions deal mainly with on line building of graphic aids to improve perception, updating the geometric database of the robotic site, and video control of the robot. A task planning module consists of a grasping configuration and trajectory generation sub-modules. This functional architecture is described in references 4 & 5. The superimposition of the real and virtual worlds is carried out through a calibration of the multisensor system; models of the sensors and several calibration methods are presented in reference 6.

The Man Machine interface we are working on is intended for performing on line modelling tasks even in case of degraded video images. This task is performed thanks to a cooperation between human operator and the sensing system, which consists of the video camera and the range finder.

Two kinds of objects are distinguished: unknown objects whose geometric data relating to the shape and the location are not known, and known objects whose geometric model is known but graphical representations of which don't match with the real object (virtual and real worlds are not superimposed). We first detail several methods used to model unknown objects and then methods to determine the pose of known ones.

II. THE PROBLEM

The remote site where the robotic task has to be performed is partially known, a 3D geometric database is set up. Human operator who operates the robot may have to face two problems: the incompletion of the geometric database, and/or the inaccuracy of data to perform the task. Two kinds of objects are distinguished: the unknown objects and the known ones.

– The unknown ones whose geometric data relating to the shape and location are not known. The operator detects such an object with the video image. A model is built: it is an embrodying volume of the object whose geometric data are determined.

– The known ones whose geometric data are already in the database. The position and orientation of an object's model in the 3D database is known with insufficient accuracy: the computer-generated and video images of this object are not superimposed. The human operator needs to correct these data in order to reach the required accuracy. The variation of the position-orientation is supposed to be small.

– The known ones whose shape are known but which are not yet set in the database. The sizes of these objects are given by the human operator. In this case in general, the gap between the visual aid and camera image can be large.

The system has to determine the position-orientation of these objects in the remote environment. The

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

representation of objects is the surface boundary representation: objects are defined by their boundaries (vertices, edges, faces). The objects' location is defined by the position-orientation of an object-referenced coordinates frame with respect to the work reference coordinates system.

III. MODELLING UNKNOWN OBJECTS

This function achieves the fusion of data provided by the operator and by sensors in order to create and set a model of an unknown object in the geometric 3D database. Qualitative information is given by the operator while quantitative information is obtained from sensors.[7] Data provided by the operator are:

– semantic: general shape (cone, polyhedron, pyramid, etc.)

– geometric: vertical, oblique, etc.

– topological: situation of the object in the environment (''vertical cone lies on the table'').

These data allow the building of an embodying volume for the unknown object. Data provided by the exteroceptive sensors:

These numerical data provided by the video camera and the range finder are used to determine the size and location of the embodying volume.

Methods concerning the modelling of conical and pyramidal volumes are presented. Thanks to the video image, the interpretation planes tangent to the object and running through the optical centre can be determined (see **Figure 1**). Using the range finder data, the geometric characteristics of the object contained in the volume bounded by the interpretation planes, are determined.

In next sections, modelling polyhedral and conical objects is presented. For each kind of objects, at least one method is presented. Besides, some experiments are reported for each method. The calibration bench presented in **Appendix 1** is used to perform these experiments.

III.1. Modelling polyhedral volumes

The general idea to model such a volume is to determine the faces' characteristics. For example, let us model the ABCD's principal face (**Figure 2**): the video camera allows the determination of the interpretation planes

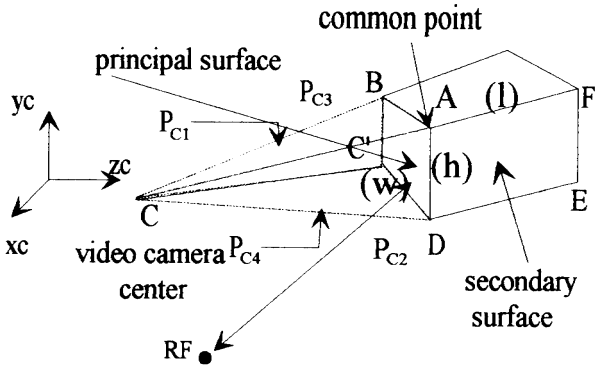


Fig. 2. Polyhedral volume modelling.

$(P_{C1}, \ldots, P_{C4})$ which contain the edges AB, BC′, C′D, DA relating to this face. By pointing a set of points belonging to this face, the range finder allows the equation of the face plane to be determined. The intersections of the face plane with the interpretation planes give the edge equations in the 3D space.

For a face which is not parallel to the z axis, the relating plane equation is: $ax + by + z + d = 0$. So for n points we also obtain a linear equation which can be solved using least-squares method:

$$A \cdot X = B \quad \text{with:} \quad A = \begin{pmatrix} x_1 & y_1 & 1 \\ x_i & y_i & 1 \\ x_n & y_n & 1 \end{pmatrix}$$

$$X = \begin{pmatrix} a \\ b \\ d \end{pmatrix} \quad \text{and} \quad B = \begin{pmatrix} -z_1 \\ -z_i \\ -z_n \end{pmatrix}$$

The same process is used to model the other faces (e.g., secondary face ADEF). Experimental results are given in **Figure 3**.

In the results presented, the relative errors in height, width, length and depth are, respectively, smaller than 3.5%, 7.5%, 4% and 5.5‰ and absolute errors respectively less than 0.16 cm, 0.36 cm, 0.29 cm and 0.35 cm. Experimental conditions are presented in appendix 1.

For the small sensor/object distance (less than 1 m) errors are greater because of the video camera/range finder parallax.

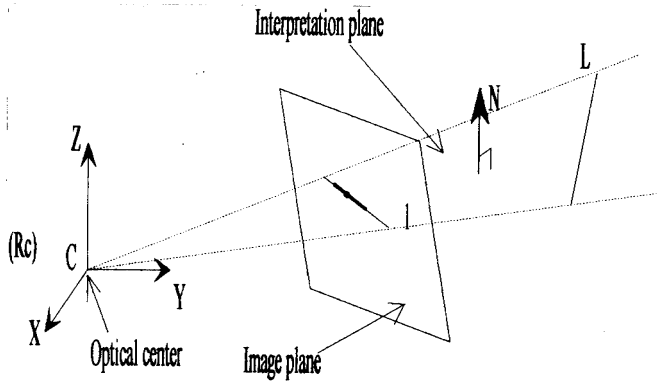


Fig. 1. Interpretation plane.

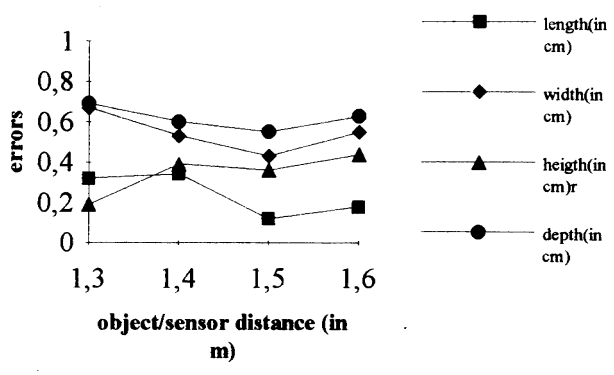


Fig. 3. Errors on polyhedral volumes.

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

In the case of our experiment, the more an object is close to the sensors, the greater is the angle between the surface normal and the laser beam. As shown in §III.2, Figure 4, that increases the measurement error.

III.2. Methods for modelling a cone

Several modelling methods using either a single sensor (range finder) or two sensors (video camera and range finder) in cooperation are presented. We determine the position/orientation of the cone's axis, as well as two radii (in the case of conic object). **Figure 4** shows the cone's profile as obtained by the range finder: we can see the increase of the error according to the angle between the surface's normal and the laser beam due to the width of the beam.

Thus, with such a range finder, it is not possible to extract directly the profile from these measures. Accurate data from the range finder are the three tangent lines to the profile: the first one is perpendicular to the range finder line of sight running through the cone axis (it is determined using selected points of **Figure 4**); the two others are lateral to the object.

By performing two profile measures with the range finder, for two different heights, it is possible to determine three tangential planes to the cone: two lateral ones (P_{RF1}, P_{RF2}) and a frontal plane (P_{RF3}) (**Figure 5).**

On the other hand, the video camera allows to determine two interpretation planes (P_{C1}, P_{C2}) relating to the lateral generating lines of the cone. Different modelling methods using at least three of these different planes can be implemented.

III.2.1. Video camera/range finder cooperation: tangential planes method

The video camera allows to determine the equations of the two interpretation planes Pc_1 and Pc_2 relative to the lateral generating lines (BC′) and (AD) of the cone (**Figure 6**).

Figure 4 shows the cone profile as obtained by the range finder. Two horizontal scannings are done with the range finder, for two different heights, near the line of sight axis which runs through the cone axis (location estimated). The equation of a plane P_{RF3} tangent to the object is determined by a least squares method. With the three distinct planes tangent to the object (Pc_1, Pc_2, P_{RF3}), its axis is determined by expressing that it is the line containing two points C_1 and C_2 equidistant from each of the three planes (**Figure 6**).

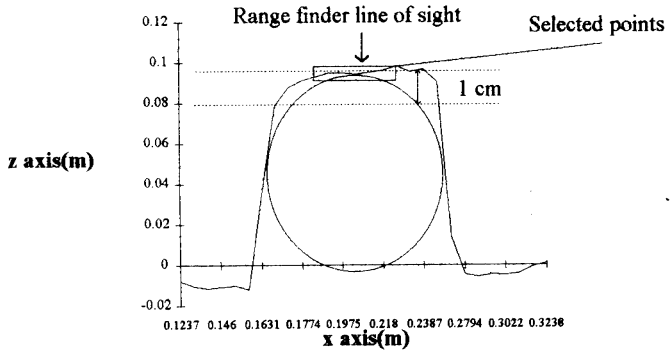


Fig. 4. Cone profile.

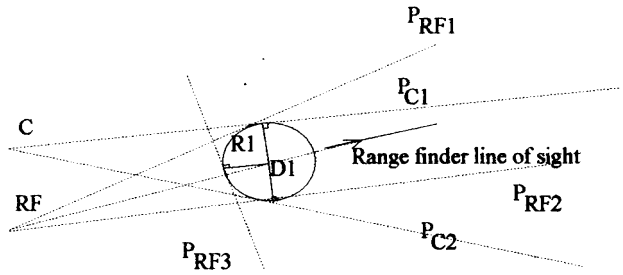


Fig. 5. Range finder tangential planes and Video Camera interpretation planes.

Each plane is defined by an equation:

$$\text{Pi: } a_i x + b_i y + c_i z + d_i = 0 \text{ with } i = (\text{c1}, \text{c2}, \text{RF3}).$$

We first express the distance D_1 between C_1 and each of the three planes by a first set of three equations:

$$\frac{|a_i x_{c1} + b_i y_{c1} + c_i z_{c1} + d_i|}{\sqrt{a_i^2 + b_i^2 + c_i^2}} = D_1 \text{ with } i(\text{c1}, \text{c2}, \text{RF3})$$

One coordinate can thus be set a priori, for example y_{c1} corresponding to a quasi vertical axis (the unknown parameters are then x_{c1}, z_{c1}, D_1), or x_{c1} (quasi horizontal axis). According to geometric considerations, to remove absolute value, let's set:

$$A = \begin{pmatrix} a_1 & c_1 & -\sqrt{a_1^2 + b_1^2 + c_1^2} \\ a_2 & c_2 & -\sqrt{a_2^2 + b_2^2 + c_2^2} \\ a_3 & c_3 & -\sqrt{a_3^2 + b_3^2 + c_3^2} \end{pmatrix}$$

$$X = \begin{pmatrix} x_{cl} \\ z_{cl} \\ D_1 \end{pmatrix}$$

and

$$B = \begin{pmatrix} -b_1 y_{c1} - d_1 \\ -b_2 y_{c1} - d_2 \\ -b_3 y_{c1} - d_3 \end{pmatrix}$$

leading to a matrix expression of the system: $AX = B$. The solution is then given by $X = (A'A)^{-1}A'B$, which determines the unknown parameters x_{c1}, z_{c1}, D_1. The

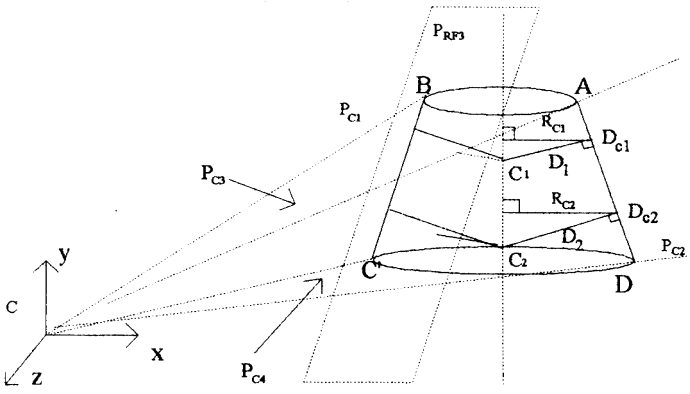


Fig. 6. Tangential planes method.

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

same process is applied with the second point C_2 for the definition of the parameters x_{c2}, z_{c2}, D_2, with y_{c2} being set. The two points' coordinates (x_{c1}, y_{c1}, z_{c1}) and (x_{c2}, y_{c2}, z_{c2}) allow now the determination of the axis equation. The projections of the two points D_{c1} and D_{c2} shown in **Figure 6** on the object's axis define respectively the two radii R_{c1} and R_{c2}, allowing the determination of all the object's primitives. In the case of a cylinder the two radii are identical. Then one can define the cone's generating line. Their intersection with two interpretation planes P_{c3} and P_{c4}, respectively defined by the AB and C′D segments, leads to the determination of the cone height (**Figure. 6**).

Results shown in **Figure 7** give an relative error less than 2% on the radius, 8‰ on the height and 5.5‰ on the depth. The absolute errors in radius, height, and depth are, respectively, smaller than 0.03 cm, 0.01 cm and 0.15 cm.

III.2.2. Range finder alone: tangential planes method

This approach uses again the process presented in the tangential planes method, but the set of the three planes needed is obtained by range finder measures only. We use the tangential lines lateral to the object to define two lateral planes P_{RF1} and P_{RF2}. The third one is P_{RF3}. The axis equation is obtained expressing the equidistance, and the radius and height are determined by the process presented previously (§III.2.1).

Errors on radius and height are comparable to the precedent method, but errors on depth are less than 6.5‰.

III.2.3. Video camera/range finder cooperation: bisecting planes method

The interpretation planes P_{C1} and P_{C2} provided by the video camera are used to determine the bisecting plane P_{MC} (**Figure 8**).

Range finder lateral and tangential planes P_{RF1} and P_{RF2} lead to the equation setting of a second bisecting plane P_{MRF}. The intersection of the two planes P_{MC} and P_{MRF} determines the equation of the object's axis.

The determination of two radii and of the primitive height is carried out by the process explained in the tangential planes method. Errors on radius and height are comparable to those of precedent methods. The depth errors are greater than in precedent methods but

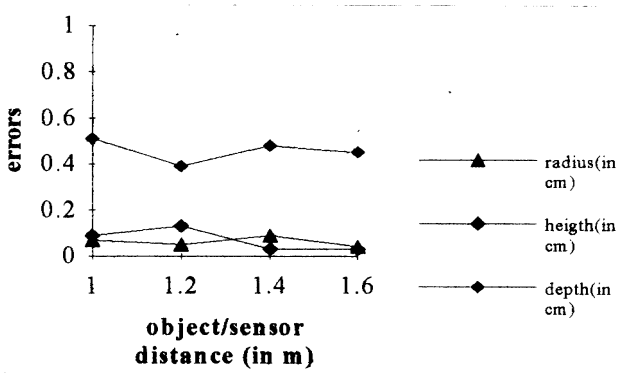


Fig. 7. Tangential planes method results.

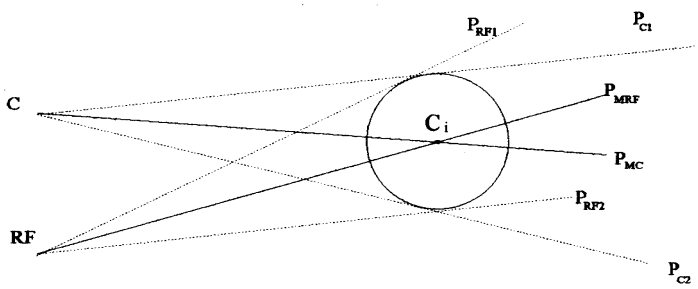


Fig. 8. Bisecting planes method.

less than 1.5%. However the depth error increase considerably when the sensor/object distance increases.

Figure 9 gives results in depth for the three methods. The one using interpretation planes by video camera/range finder cooperation seems to have the best results. When the object-sensor distance increases the bisecting planes method error increases rapidly. In fact, when the object move away, the angle between the two bisecting planes is small. So an error on the determination of one the plane introduces an error on the position of the intersection of the two planes.

Figure 10 shows the result of superimposition of the real and virtual worlds on a parallelipiped and a cone volume.

IV. LOCATION DETERMINATION OF KNOWN OBJECTS USING SINGLE CAMERA VIEW

In this process, 3D data of the object are known in the world coordinates frame (Ro). When the object moves, the virtual and real worlds aren't superimposed. So, it is necessary to update 3D database. Human operator points out, onto the screen, 2D characteristics of the object's image. So, the matching between 3D and 2D characteristics of the object is performed by the human operator.

The method allows to determine geometrical transformations representing object motion.

The equation of each edge Li of the object is known with respect to a coordinate frame (Ro) attached to the object. The process must determine the position-orientation (Rm) of (Ro) in the environment, i.e. the rotation matrix R and the translation vector T defining the coordinates transformation between (Ro) and (Rm)

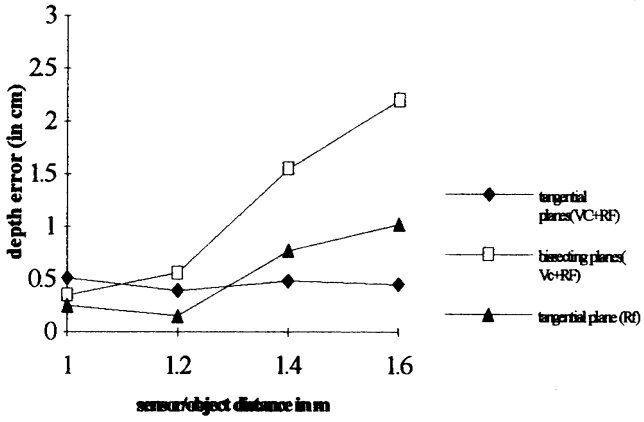


Fig. 9. Depth error for the three methods.

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

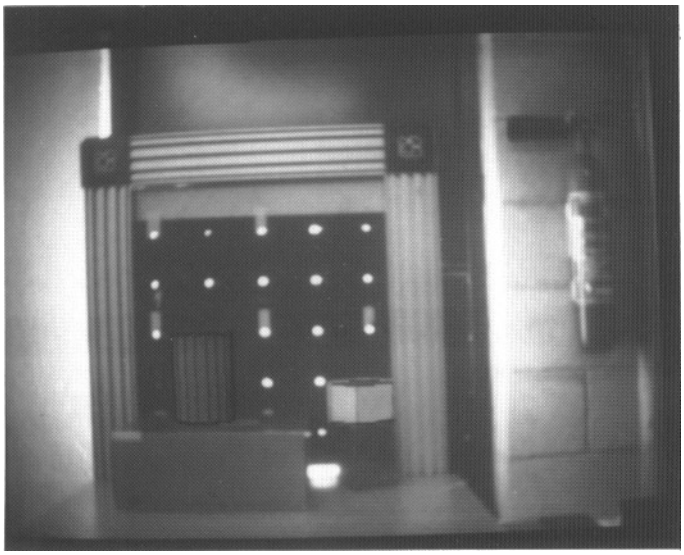


Fig. 10. Superimposition of the real and virtual worlds on the calibration bench.

(**Figure 11**). The visual marks used by the human operator are the image li of edge Li.

Let Mo and $\vec{V}o$ be the coordinates of a point and vector of the object expressed with respect to (Ro). M and $\vec{V}$ are the new coordinates of Mo and $\vec{V}o$ after the rotation R and translation T. Thus:

$$M = T + RMo \qquad \vec{V} = R.\vec{V}o$$

The visual mark $\vec{li}$ with the associated vector $\vec{vi}$ and the point mi (image of Mi) allow the determination of the interpretation plane defined with the normal vector $\vec{Ni}$ and containing the edge Li and the associated vector $\vec{Vi}$ (**Figure 12**).

$$\vec{Ni} = \vec{vi} \times \vec{cm}\ i$$

where ($\times$) is the cross product

So R and T satisfy the two relations:

$$\vec{Ni}.(R.\vec{Voi}) = 0 \quad (1)$$

$$\vec{Ni}.(T + RMoi) = 0 \quad (2)$$

where (.) is the dot product.

Three edges Li ($i = 1, 2, 3$) are used to solve the first equal to determine R, then to find T using the second equation.

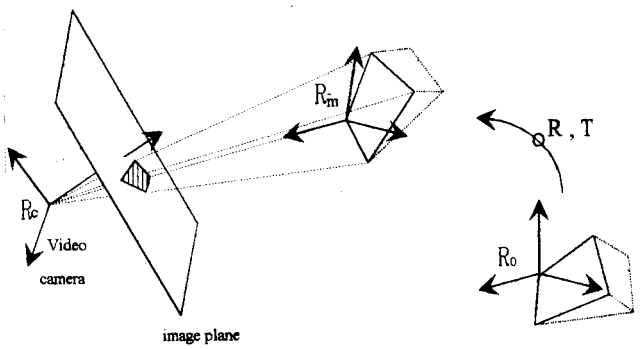


Fig. 11. Position-orientation of a known object.

Many solutions have been proposed to solve this problem References 8, 9 have a geometric approach; References 10–12 have a numerical approach. All these approaches are heavy in processing time.

The direct resolution of the orientation from (Rm) to (Ro) is complex. In order to reduce computation complexity, we introduce intermediary coordinate frames.

In the next sections, we present the two steps of the method, the first one is used in case of object large motion and the second one in case of object small motion.

These algorithms can be used successively, the first one provides an estimation of the pose and the second one improves the accuracy obtained in the first case.

IV.1. Case of large amplitude

This method is based on intermediary reference coordinate systems definition which simplify problem resolution, and rotation, translation determination which give the 3D object attitude.

IV.1.1. The different reference coordinate systems

The reference coordinate systems used are the following (**Figure 13**):

❶ (Ro) = object reference coordinates,
❷ (Rc) = camera reference coordinates
❸ (Ra) = viewing reference coordinates,

We define the following matrices:

↪ $C_{3\times4}$ = camera global model, projection matrix from (Ro) to (Ra).
↪ Toc = camera external model, homogeneous transformation matrix from (Ro) to (Rc).
↪ Mint = camera internal model, matrix defining the coordinate transformation from (Rc) to (Ra).
❹ (Rm) = reference frame attached to the object. In initial position, (Rm) is equal to (Ro).
❺ (Rm1) = reference defined with the current 3D edge of the object (see §IV.1.3.2).
❻ (Rc1) = reference defined with 2D data of the object (see §IV.1.3.1).

We define, then the following matrices:

↪ Roc1 = rotation matrix from (Ro) to (Rc1).
↪ Rmm1 = rotation matrix from (Rm) to (Rm1).
↪ (R, T) = transformation matrices from (Ro) to (Rm).
↪ Rc1m1 = rotation matrix from (Rc1) to (Rm1).

The method consists in finding the rotation R and the translation T from (Ro) to (Rm) through a geometrical approach. It is based on DHOME's work.[8] The direct resolution introduces the three Euler angles and gives complex equations not easy to solve.

In order to reduce complexity, Dhome introduces two coordinate frames Rc1 and Rm1, in such a way that Rc1m1 depends on two angles only: so it is possible to determine the solution.

IV.1.2. Contribution of our approach

The method of large amplitude is based on DHOME's. However DHOME works with the internal and external camera models separately. His method is based on

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

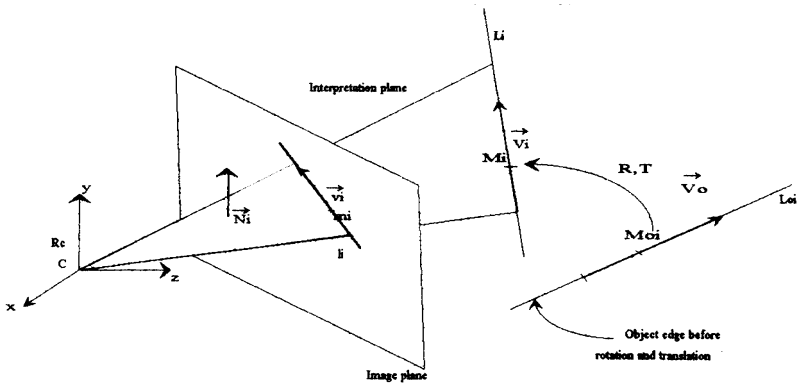


Fig. 12. Elements of the position-orientation determination.

internal parameters of the camera and after each object's motion calculates the external camera model (transformation matrix from (Rm) to (Rc)).

The originality of our approach consists in working with the global camera model It both reduces the number of calculation and improves the accuracy of the method.

Due to the calibration methods of cameras, the global model which gives relation between pixel coordinates and 3D coordinates of points is always good. But internal and external parameters determined either from the global model [6] or directly from an optimisation process, may sometimes be unstable. In the latter case, we use Levenberg-Marquardt algorithm which can sometimes converge to a slightly wrong value of parameters.[13]

Besides, to solve equation mentioned §IV.1.3.3, a numerical method, faster than the iterative algorithm used by Dhome,[8] is used. This method is described in appendix 2.

IV.1.3. Determination of the rotation R from (Ro) to (Rm)

First let us define the matrices used.

IV.1.3.1. Roc1 determination, rotation matrix from (Ro) to (Rc1). Camera global model $C_{3\times4}$ expresses the relationship between the 2D coordinates $p(u, v)$ of the image of a given 3D point

$$P\begin{pmatrix} X \\ Y \\ Z \end{pmatrix}$$

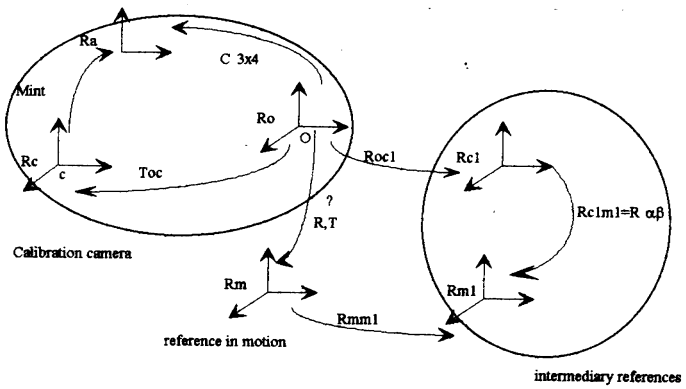


Fig. 13. The different reference coordinate frames.

in the reference (Ro), we have:

$$\begin{pmatrix} s*u \\ s*v \\ s \end{pmatrix} = C3\times4.\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \tag{3}$$

with:

$$C_{3\times4} = \begin{pmatrix} C11 & C12 & C13 & C14 \\ C21 & C22 & C23 & C24 \\ C31 & C32 & C33 & C34 \end{pmatrix}$$

The development of relation (3) gives two independent equations:

$$\begin{cases} nu.P + au = 0 \\ nv.P + av = 0 \end{cases} \tag{4}$$

where

$$P = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}, \quad nu = \begin{pmatrix} C11 - u*C31 \\ C12 - u*C32 \\ C13 - u*C33 \end{pmatrix},$$

$$nv = \begin{pmatrix} C21 - v*C31 \\ C22 - v*C32 \\ C23 - v*C33 \end{pmatrix}, \quad au = (C14 - u*C34)$$

and

$$av = (C24 - v*C34)$$

Relation (4) represents two plane equations. The cross product of the normal vectors to the two planes (nu and nv) gives the visual ray running through P. Visual ray is a line defined by camera optical center(c) and (2D pointed out point(p). The human operator uses the image li (i = 1 to 3) of the edge Li (i = 1 to 3) as visual marks. The visual ray of the first 2D vertex of l1 defines the $\vec{x}$ axis of (Rc1). So, the $\vec{x}$ axis of (Rc1) is in the interpretation plane of l1.

The cross product of $\vec{x}$ axis and visual ray of the second 2D vertex of l1 defines the interpretation plane. This normal vector of the interpretation plane given by l1 defines the $\vec{y}$ axis of (Rc1).

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

We define $\vec{z}$ axis by the cross product of $\vec{x}$ and $\vec{y}$. $(\vec{x}, \vec{y}, \vec{z})$ defines the rotation matrix from (Ro) to (Rc1): the coordinates of $(\vec{x}, \vec{y}, \vec{z})$ form the columns of the matrix Roc1 or the lines of the inverse matrix Rc1o.

Only l1 image of L1 is used to determine Roc1.

IV.1.3.2. Rmm1 determination, rotation matrix from (Rm) to (Rm1). Li (i = 1 to 3) are the edges of the current position of the object. The initial coordinates of Li in (Ro) are the same as the coordinates of Li in (Rm).

The $\vec{x}$ axis of (Rm1) is carried by L1. So, the $\vec{x}$ axis of (Rm1) is in the interpretation plane of l1. The $\vec{z}$ axis of (Rm1) is given by the cross product of $\vec{V}1$ and $\vec{V}2$ respectively carried by L1 and L2. We define the $\vec{y}$ axis by the cross product of $\vec{z}$ and $\vec{x}$. $(\vec{x}, \vec{y}, \vec{z})$ defines the rotation matrix from (Rm) to (Rm1). The coordinates of $(\vec{x}, \vec{y}, \vec{z})$ form the columns of the matrix Rmm1 or the lines of the inverse matrix Rm1m. Only L1 and L2 are used to define Rm1m.

IV.1.3.3. Rc1m1 determination, rotation matrix from (Rc1) to (Rm1). We have seen that both (Rc1) and (Rm1) $\vec{x}$ axis are in the interpretation plane of l1. Also, the $\vec{y}$ axis of (Rc1) is the normal vector of the interpretation plane of l1. Therefore, the rotation matrix from (Rc1) to (Rm1) depends on two rotations only. The rotation $(\vec{y}, \beta)$ so that the two $\vec{x}$ axis become parallel and the rotation $(\vec{x}, \alpha)$ so that the two $\vec{y}$ axis become parallel. If these correspondences are achieved, the two $\vec{z}$ axis become automatically parallel.

Now the rotation matrix from (Ro) to (Rm) is given by:

$$\mathbf{R = Roc1\ Rc1m1\ Rm1m}$$

So, the determination of R consists in finding Rc1m1 which depends on unknown angles (α, β). We obtain a polynomial equation of degree 8 (see Appendix 2). The computation gives α and β and thus Rc1m1.

Equations are set in form with symbolic calculation software, whose syntax is compatible with C language. It permits the integration into C program. Polynomial equation of degree 8 is obtained. It is computed with a numerical method based on the determination of the eigenvalue of the companion matrix.

IV.1.3.4. Determination of the translation T from (Ro) to (Rm). The translation is given by equation:

$$\overrightarrow{\mathrm{Ni}}/\mathrm{Ro}.\mathrm{c}\overrightarrow{\mathrm{Mi}}/\mathrm{Ro} = 0$$

$$\Leftrightarrow \overrightarrow{\mathrm{Ni}}/\mathrm{Ro}.(\mathrm{T} + \mathrm{RMi}/\mathrm{Rm} - \mathrm{c}/\mathrm{Ro}) = 0 \quad (5)$$

where:

Mi/Rm is a 3D point of Li, c/Ro is the center of (Rc)

Note:

The calculation of the translation T depends on the rotation R. The translation includes three unknown parameters (a, b, c). Edges Li (i = 1 to 3) and points Mi (i = 1 to 3) are used for translation determination (where each point Mi belongs to the edge Li).

The edges Li (i = 1 to 3) used to determine the rotation R are triangular. This configuration gives 4 solutions in rotation. The solutions (1′) and (2′) are rejected because they are not in the direction of the camera view. A test of distances between 2D points of image obtained and initial image allows the choice of the good solution (1) (**Figure 14**).

However, to determine the translation T, the three image lines li of edges Li must not be convergent.

This method is tested thanks to an experiment which is described in section VI.

IV.2. Case of small amplitude

In this case, we have implemented and tested an original method. With this method, $\vec{R} = \theta\vec{r}$ is expressed as a function of the vector rotation $\vec{r}$, which represents the axis and angle θ of the rotation. This method consists in linearizing the motion with respect to the three coordinates of $\vec{r}$. It is available to rectify the orientation of an object in order to get the computer-generated and video images rigorously superimposed. For small rotations:

$$\vec{V}' = \vec{V} + \vec{R} \times \vec{V} \quad \text{where:} \quad \vec{V}' = \vec{V} + d\vec{V}$$

with:

$$d\vec{V} = \vec{R} \times \vec{V} \quad (\times \text{ cross product}) \qquad (6)$$

$\vec{V}$: vector director of a 3D edge of the object in initial position, $\vec{V}'$ vector director of the same edge after motion.

In equation (6) the translation doesn't appear, so we introduce the translation by reducing the expression to points, we have:

$$dP = T + \vec{R} \times \overrightarrow{OP}$$

O is the center of (Ro), so:

$$dP = T + \vec{R} \times \vec{P} \qquad (7)$$

The derivation of the equations (4) (case of small motion: rotation and translation) gives:

$$\begin{cases} dnu.P + nu.dP + dau = 0 \\ dnv.P + nv.dP + dav = 0 \end{cases} \qquad (8)$$

dP is given by equation (7), we introduce the transformation:

$$nu.(R \times p) = (P \times nu).R$$
$$nv.(R \times p) = (P \times nv).R$$

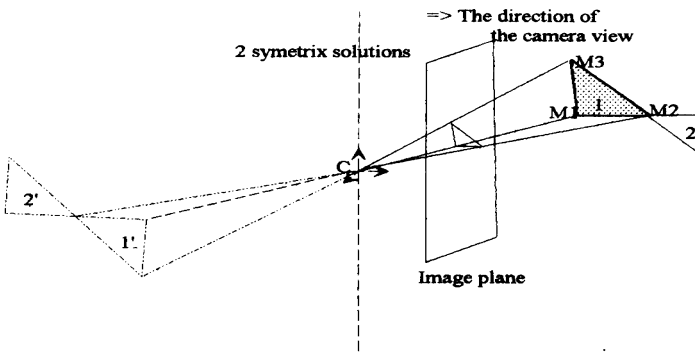


Fig. 14. The possible solutions.

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

we obtain:

$$\begin{cases} nu.T + (P \times nu).R + dnu.P + dau = 0 \\ nv.T + (P \times nv).R + dnv.P + dav = 0 \end{cases} \quad (9)$$

which can be written as a matrix:

$$AX = B \quad (10)$$

where: $X = (a, b, c, Rx, Ry, Rz)^t$ and $T = (a, b, c)^t$, $R = (Rx, Ry, Rz)^t$

The equation (10) can be solved by using the least squares method, if more than three points P are taken.

IV.3. Results

The algorithm assessment is carried out first with a simulation process and then with measurements on real objects.

Object tested: parallelepiped (height = 13.2 cm, width = 23.1 cm, length = 35 cm). Average distance-camera/object: 2m

We have two kinds of tests:

In the first test the 2D coordinates are given by the computer. In this test there are no noise measurements so the result gives the accuracy of the method.

In the case of large amplitude the algorithm works with $(10^{-6})°$ errors for the rotation and 10^{-6} m errors in translation (**Figure 15a and b**).

In the case of small amplitude, we have tested the validity domain of the small motion. The algorithm is accurate for a rotation which is not greater than 18° and the translation less than 5 cm. The algorithm is considered accurate when the error in the rotation angle value is less than 0.5° (**Figure 16**).

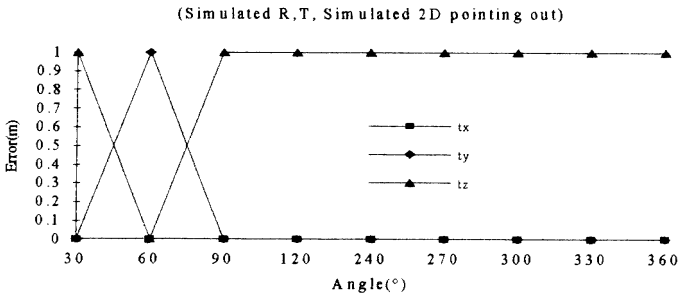


Fig. 15a. Error in translation ($\times 10^{-6}$ m) for the large amplitude method.

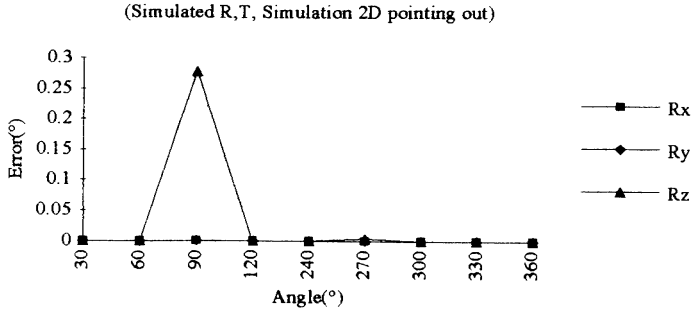


Fig. 15b. Error in rotation ($\times 10^{-6°}$) for large amplitude method.

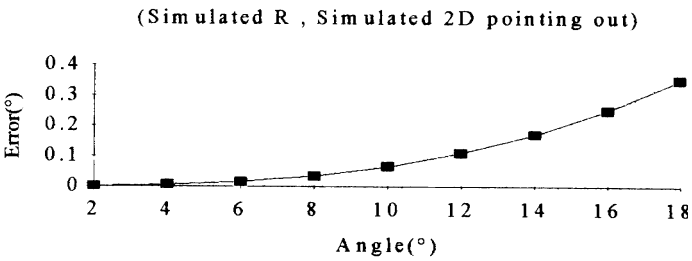


Fig. 16. Validity of the small amplitude method.

In the second test the human operator points out the 2D points on the video image. We have applied successively the two algorithms. The results are presented (**Figure 17a and b**). The algorithm of large amplitude gives error less than 1° in rotation, but the error in translation is not small, up to 4 cm. The method of small amplitude gives an error less than 0.3° in rotation and less than 1 cm in translation (**Figure 17a and b**).

The use of redundant data in the second step provides a better accuracy and reduces standard derivation of the error (**Table I**). The algorithm of large amplitude gives 1.4 cm and 1.3 cm of mean value and standard deviation of error in translation, and 0.23° and 0.19° in rotation. Therefore the method of small amplitude gives 6 mm and 2 mm of mean value and standard deviation of error in translation, and 0.1° and 0.1° in rotation.

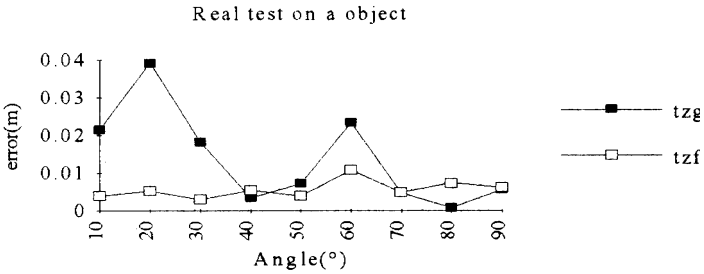


Fig. 17a. Error in translation on Z axis for the two methods.

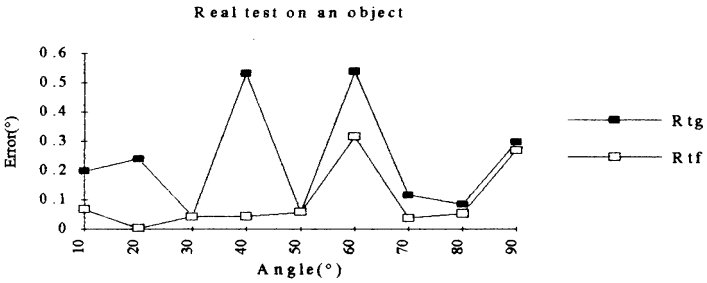


Fig. 17b. Error in rotation for the two methods.

Table I
Mean value and standard deviation of the errors

	Mean (g)	Mean (f)	Stan. dev. (g)	Stand. dev. (f)
tz(m)	0.01378	0.00563	0.01268	0.00233
R(°)	0.23342	0.09810	0.19033	0.11136

g: for large amplitude method, f: for small amplitude method

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press



Fig. 18. Superimposition of real and virtual worlds on real site.

V. EXPERIENCE ON REAL SITE

A pick and place task is performed as a validation of this work. The task is decomposed into four steps and took into account the robot MAT2 we have in the laboratory.

– Modelling object, the position-orientation of the cylinder (*dimensions:* $r = 5.1$ cm, $h = 16.6$ cm) is defined by the modelling conical method described in (§III.2.1). **Figure 18** shows the superimposition of the real and virtual worlds.

– Pick object: the object is pointed out in the database by pointing out its 2D image on the video camera. The geometric information on the object position is transmitted to the robot which comes to pick up the cylinder.

– Put object: the robot puts the object down on the place pointed out, by human operator, on the table. The cylinder is on a table (**Figure 19**). The equation plane of the table is given by the pose face of the cone. Thus, the determination of the placing point is given by pointing out its 2D image on the video camera.

– Database updating: the database is update by means of the proprioceptive sensors of the robot.

VI. CONCLUSION

Methods for building and updating an environment's 3D geometric database in telerobotics have been described in this paper.

The operator models unknown objects (of conical or polyhedral pattern) using a video image and a range finder. The improvement offered by the cooperation video camera/range finder has been shown for small depths (below 1.6 m). In the case of conical objects several modelling methods have been presented and assessed.

The location determination method was tested when the dimensions of objects are known. It is composed of two steps which are applied successively. In the first step, when the motion of the object is large, a geometric method is used. An approximate solution is obtained. In the second step, the case of small motion of an object, we use an algorithm which improves the accuracy obtained

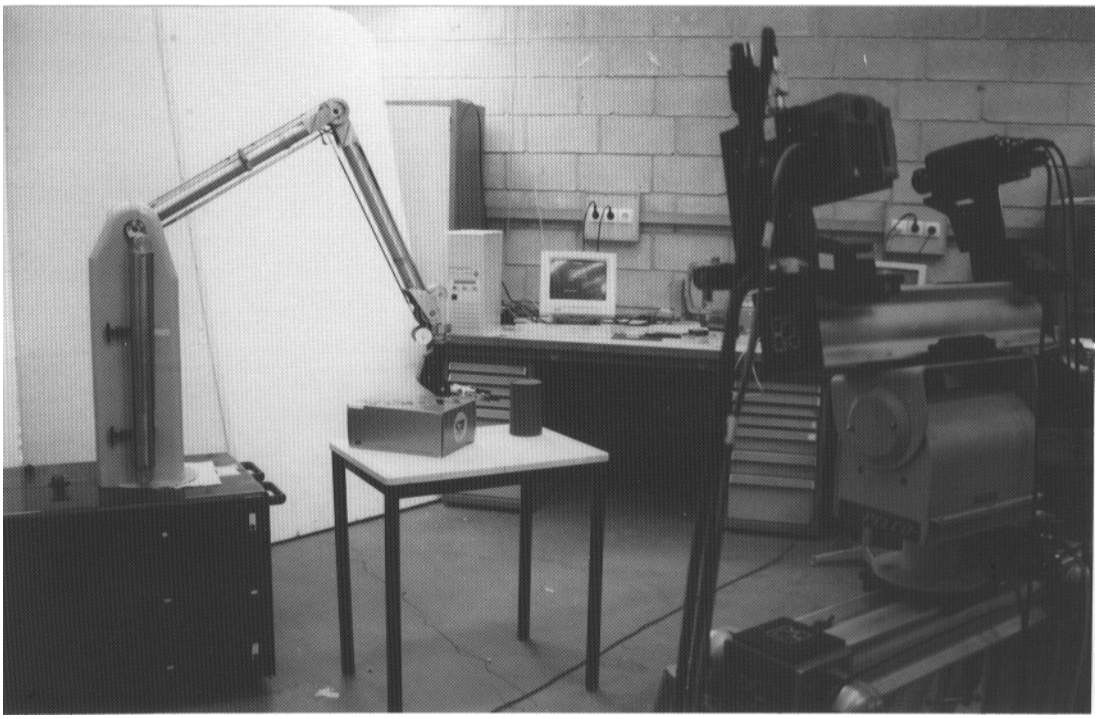


Fig. 19. Pick task of a cylinder.

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

in the first step. The tests for the algorithm accuracy were carried out first by simulation and then in real case.

For the experiment on real object the 2D image coordinates are pointed out by the human operator. The noise measurement reduces the accuracy of the large amplitude method. The error in rotation is however less than 1°, but the error in translation is not small, up to 4 cm. Applying the method of small amplitude, the error in rotation becomes less than 0.3° and in translation less than 1 cm.

The algorithms are accurate for an additional noise smaller than 2 pixels.

These methods are tested on a real telerobotics environment. Results show that accuracy obtained is sufficient for telerobotics task performing.

The future research consists in the automation of updating database procedure. It consists in the matching between 2D image and its 3D model.

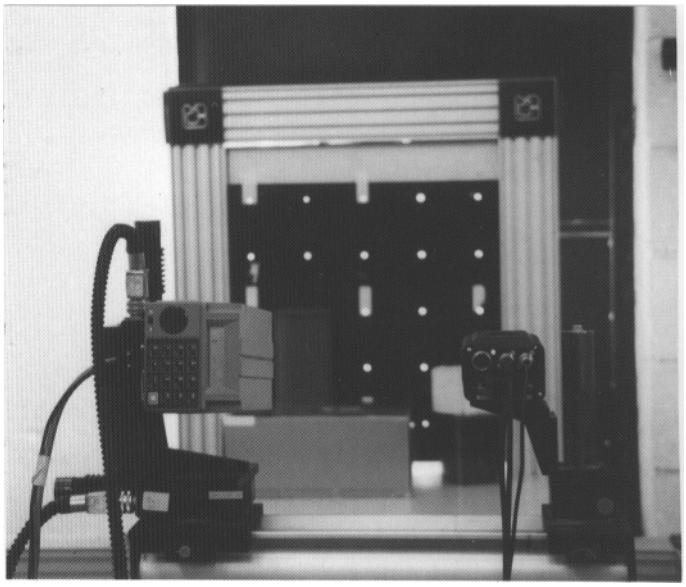
Reference

1. T.B. Sheridan, "Telerobotics" *Automatica,* Vol. 25, no. 4, 1989; pp. 487–507.
2. Ph. Even, and R. Fournier, "Telerobotics task execution based on 3D geometric modelling and graphical programming." *IEEE/SMC'93, Systems, Man and Cybernetics Le Touquet France* (1993) **Vol. 3,** pp. 132–37.
3. T. Hasegawa, T. Suehiro, T. Ogasawara, T. Matsui, K. Kitagari and K. Tagase, "A Model-based Tele-robot System with Manipulation Skills" *ISART* Tokyo, Japan, (March, 1991) pp. 499–506.
4. M. Mallem, E. Colle and F. Chavand, "A Multimedia Control Interface in Teleoperation" *IEEE/SMC'93 Conference. System Engineering in the Service of Humans.* Le Touquet, France (October 17–20, 1993) **Vol. 3,** pp. 126–131.
5. M. Mallem, F. Chavand and E. Colle, "Computer-assisted visual perception in teleoperated robotics" *Robotica* **10,** Part 1, 93–103 (1992).
6. Y. Chekhar, E. Colle, F. Chavand and A. Loukil, *Calibration multicapteur d'un systeme d'aide a la téléopération* (APII, Hermès. France, 1994) **Vol 28,** No. 6, pp. 599–623.
7. M. Koizumi and F. Tomita, "Qualitative and quantitative of solid models and images of 3D objects" *Proc. 9th Int. Conf. on Pattern Recognition* (Nov., 1988) pp. 681–684.
8. M. Dhome, M. Richetin, J. T. Lapreste and G. Rives, "Determination of the attitude of 3D objects from a single perspective view" *IEEE Transactions on Pattern Analysis and Machine Intelligence* **11,** No. 12, 1265–1278 (Dec. 1989).
9. R. Horaud "New methods for Matching 3D objects with single perspective views" *IEEE Transactions on Pattern Analysis and Machine Intelligence* **PAMI-9**. No. 3. 401–412 (May, 1987).
10. J. S. C. Yuan, "A general photogrammetric method for determining object position and orientation" *IEEE Transaction on Robotics and Automation,* **5**(2), 129–142 (April, 1989).
11. D. Lowe, "Three-dimensional Object Recognition from Single Two-dimensional Images" *Artificial Intelligence* **31,** 355–395 (1987).
12. R. Horaud and O. Monga. *Vision par Ordinateur, Outils Fondamentaux* (Hermes, Paris, France, 1993).
13. J. Triboulet and F. Chavand, "Calibration of a Multisensor System" *CESA'96 IMACS/IEEE-SMC Computational Engineering in Systems Applications,* Lille, France (July 9–12, 1996) pp. 664–669.

Appendix 1: The calibration bench

Sensors and the calibration bench

Camera: focal = 8.5 mm, definition = 756 × 256 photoelements, graphic card definition: 512 × 256.



Appendix 1. The calibration bench.

3D range finder:

Range finder:

statistical accuracy on distance $r = 3,\ 5,\ 10,\ 20$ mm range = 0.6–10 m.

2 dof Turret:

angular accuracy on θx and θy: 0.01°.

Appendix 2

Determination of the polynomial equation

The visual mark li gives the normal vector $\overrightarrow{\mathrm{Ni}}$ of the interpretation plane which contains the edge Li and the associated vector $\overrightarrow{\mathrm{Vi}}$. We have:

$$\overrightarrow{\mathrm{Ni}}/\mathrm{Rcl}.\overrightarrow{\mathrm{Vi}}/\mathrm{Rc1} = 0 \ (\mathrm{i} = 1 \text{ à } 3) \qquad \text{(A1)}$$

However we know $\overrightarrow{\mathrm{Ni}}/\mathrm{Ro}$ and $\overrightarrow{\mathrm{Vi}}/\mathrm{Rm}$, so we use the different transformation matrices

$$\Rightarrow [\mathrm{Rc1o}\overrightarrow{\mathrm{Ni}}(\mathrm{Ro})].[\mathrm{Rc1m1Rm1m}\overrightarrow{\mathrm{Vi}}(\mathrm{Rm})] = 0 \qquad \text{(A2)}$$

where:

$$\mathrm{Rc1m1} = \mathrm{R}\beta\ \mathrm{R}\alpha = \begin{pmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\alpha) & -\sin(\alpha) \\ 0 & \sin(\alpha) & \cos(\alpha) \end{pmatrix}$$

The development of the dot product in the equation (A2) for (i = 1 to 3) gives three equations:

$$\mathrm{Gi}(\alpha, \beta) = 0 \quad (\mathrm{i} = 1 \text{ to } 3)$$

We used the two last equations:

$$\begin{cases} \mathrm{G2}: \mathrm{s1}(\alpha)^* \cos(\beta) + \mathrm{s2}(\alpha)^* \sin(\beta) = \mathrm{s3}(\alpha) \\ \mathrm{G3}: \mathrm{s4}(\alpha)^* \cos(\beta) + \mathrm{s5}(\alpha)^* \sin(\beta) = \mathrm{s6}(\alpha) \end{cases} \qquad \text{(A3)}$$

where si (i = 1 to 6) is a function of cos (α) and sin (α)

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press

Note:
Below, we present conditions needed to solve equation (A3).

– Two equations are necessary to find α and β.

– Both Li (i = 1 to 3) and their 2D images must be different.

– Equation G1 isn't used because it's invariable with respect to α.

– Edges Li (i = 2, 3) must not be parallel to L1.

Equations (A3) gives $\cos(\beta)$ and $\sin(\beta)$ expressed as a function of α.

We introduce: $\cos^2(\beta) + \sin^2(\beta) = 1$ and then obtain an equation:

$$F(\alpha) = 0 \tag{A4}$$

We introduce

$$\cos(\alpha) = \frac{1 - t^2}{1 + t^2} \quad \text{and} \quad \sin(\alpha) = 2^* \frac{t}{1 + t^2}$$

We obtain a polynomial equation of degree 8. The computation gives α and β and thus Rc1m1.

https://doi.org/10.1017/S026357479700060X Published online by Cambridge University Press