2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

An algorithm for optimal closed-path generation over arbitrary surfaces using uncalibrated vision*

Emilio J. González-Galván, Ambrocio Loredo-Flores
Erika D. Laborico-Avilés, Felipe Pazos-Flores
CIEP, Facultad de Ingeniería
Universidad Autónoma de San Luis Potosí
Av. Manuel Nava 8. Zona Universitaria.
San Luis Potosí , SLP, 78290, Mexico
egonzale, aloredof@uaslp.mx

J. Jesús Cervantes-Sánchez
FIMEE, Departamento de Ingeniería Mecánica
Universidad de Guanajuato
Calle Tampico No. 912, Col. Bellavista.
Salamanca, Gto., 36730, Mexico
jecer@salamanca.ugto.mx

***Abstract*— This paper presents aspects related to the generation and tracking of closed trajectories over an arbitrary surface of unknown geometry. Such a capacity is required in some of the most important industrial robotic tasks, like the cutting or welding of commercial plates. Previous work has shown how a calibration-free, vision-based method can be combined with an optimal geodesic-mapping approach, in order to generate an optimal trajectory that traces a previously defined path, stored as a CAD model, over an arbitrary, curved surface. The application of this technique in the case of non-developable surfaces did not achieve closure when a closed-trajectory was attempted. This paper presents a methodology for successfully achieving closure of a given closed-path, when this is traced over a non-developable surface. The proposed technique was tested using an industrial robot and a vision-based system that included structured lighting for image-analysis simplification. The results of these experiments are reported herein.**

***Index Terms*— Camera-space manipulation, optimal mapping, vision.**

I. INTRODUCTION

Many important robotic manufacturing tasks require that a given trajectory has to be precisely followed, over an arbitrary surface of unknown geometry. Examples of such tasks include the covering of melted metal over a worn surface [1], the cutting of arbitrary shapes over commercial plates and the cutting and welding over the surface of large recipients, required for placing attachments like flanges and man-holes. In the case of robotic laser and plasma-cutting, the precision required in the placement of the cutting-tool over the surface is about $1\ mm$ for positioning and 1^o for the orientation of the cutting-nozzle, with respect to the surface normal [2]. In these two cases, the large variation in geometry of a worn-surface or the uncertainty in geometry of a commercial plate, caused by bending, involves an intensive programming process, different for each piece. The ideas presented in this paper can be directed towards facilitating such a demanding programming process.

*This work is partially supported by the National Council for Science and Technology of Mexico (CONACYT) and the Fund for Research Support (FAI) of UASLP

The tasks listed above, specially cutting, has a fundamental requirement: When a closed trajectory is developed, closure must be guaranteed. Previous work [3] has shown a successful application of a vision-based method applied to the problem of optimal path-generation and tracking over an arbitrary surface. This method considered geodesic-mapping in order to transfer a path, stored as a CAD model, over a real, curved surface. Such a methodology was successfully developed and tested, using an industrial robot to perform the task. It also achieved a successful closure of a closed path when the testing surface was a developable one. However, in the case of non-developable surfaces, closure was not achieved. This was the result of a combination of causes. In the first place, the geodesic mapping that was used for this method is, by definition, incapable of producing closure [4] when a mapping between a virtual, flat surface (CAD model) and a non-developable surface (real surface) is sought. Additionally, the proposed methodology carried an integration process that produced an accumulation of error as the path is developed. An algorithm for solving these drawbacks is proposed herein, in the following sections.

It is important to mention that the industrial tasks cited above, when performed, produce deformations in the piece due to heat. The deformation of the piece may affect adversely the result of the manufacturing task and, since the vision-based control strategy cannot be applied while the task is performed (due to the amount of light and smoke emitted during the industrial process), one may tend to underestimate the importance of defining an optimal sequence of arm motions prior to the actual development of the industrial task. However, there exist in the market commercial devices that take these heat-induced deformations into account, for instance, the *Adaptive Welding*^{TM} tool [5], from Fanuc Robotics. Although details are not given regarding this particular tool, it can be mentioned that a laser-joint scanner is used to dynamically adjust welding parameters like weaving, travel speed, voltage and wire-feed speed to volumetrically fill the joints. In order to work, these tools require a previously-defined path, which is modified according to the deformation present on the piece. Such an original path is the one that

results by applying the methodology presented in this paper. Therefore, the proposed strategy is capable of coexisting with the commercially-available tools designed for monitoring and correcting the robot-path, according to the changing conditions in the geometry of the workpiece during task execution.

II. General description of a path-generation and tracking task.

The vision-based method proposed herein requires simultaneous path-generation and path-tracking processes. This is described schematically in the flow-diagram of Fig. 1. This diagram describes the case in which a given trajectory is followed only once.

When the program is finished, a sequence of robot configurations that enable a successful execution of the task is obtained. In this type of maneuvers, the closed trajectory is defined ahead of time in a database (CAD model). The information included in this database is depicted schematically in Fig. 2. In this case, the trajectory is divided into a sequence of straight-line segments with coordinates (r_i,θ_i,z_i), relative to the previous segment. Also, associated to each segment is the speed of the tool and a flag used to establish whether a given task, like welding, cutting, etc., is performed or not in the segment.

Prior to the development of the path-generation-and-tracking task, a set of vision parameters must first be defined. These parameters are associated to an orthographic camera-model that relates the arm configuration with the appearance, in camera-space, of a series of visual features attached to the tool of the robot. The process of estimating such parameters is presented in detail in [6].

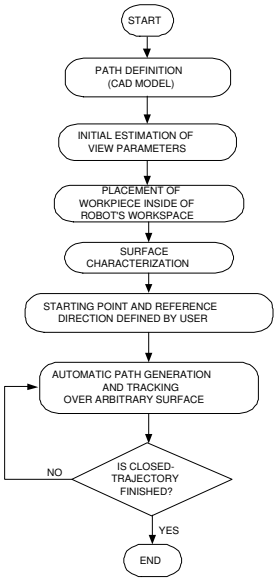

Fig. 1. Flow-diagram of a path-tracking task.

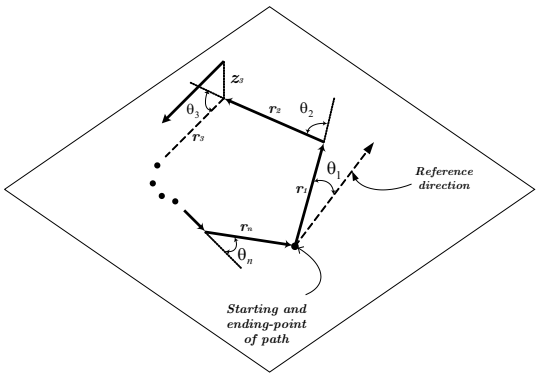

Fig. 2. Structure of information stored in a database (CAD model).

The next step consists on placing the workpiece in an arbitrary fashion, inside of the robot's workspace. The control cameras used in the maneuver point to different regions of the workpiece, as shown schematically in Fig. 3.

The following sections will provide additional details on several aspects related to the maneuver for tracing the path, presented in the flow diagram of Fig. 1.

III. Surface characterization.

The process of surface characterization involves the projection of structured lighting in the form of a matrix array of laser beams. The application of laser has, among others, the advantage of not producing a permanent marking of the surface. Additionally, simple image-analysis algorithms, like those presented in [7] and [8], can be used to detect the center of laser spots, in the images obtained from the control cameras. Fig. 4 shows a picture of the structured light used in the process of surface characterization. As presented in [3] the image-plane information gathered from the projection of structured lighting is restricted, based on ideas presented in

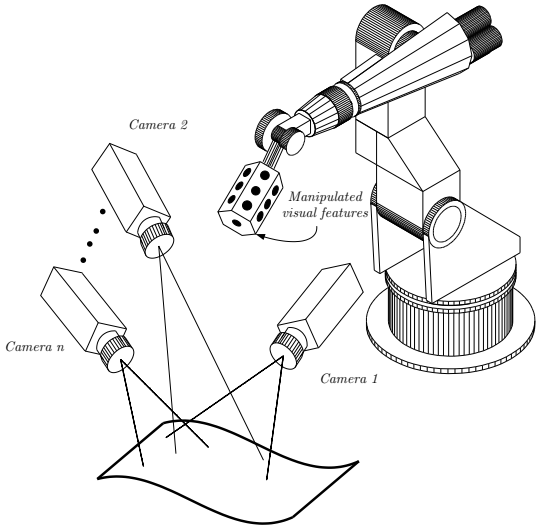

Fig. 3. Control cameras used in the maneuver.

Fig. 4. Structured lighting in the form of a matrix array of laser spots.

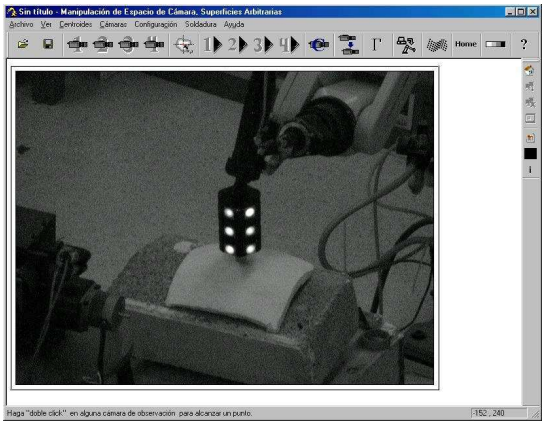

Fig. 6. Graphical user interface used in experiments.

[9]. Fig. 5 shows the detected laser spots that lie within a user-defined region, corresponding to the surface of interest.

In the experiments reported here, an orthographic camera model is used. Such a model represents the non-linear, algebraic correspondence between the three-dimensional location of the center of the laser spots and their appearance in camera-space. The model is as follows,

$$\begin{aligned} f_x(x,y,z;\mathbf{C}) &\equiv b_1(\mathbf{C})\ x\ +\ b_2(\mathbf{C})\ y \\ &\quad +\ b_3(\mathbf{C})\ z\ +\ b_4(\mathbf{C}) \\ f_y(x,y,z;\mathbf{C}) &\equiv b_5(\mathbf{C})\ x\ +\ b_6(\mathbf{C})\ y \\ &\quad +\ b_7(\mathbf{C})\ z\ +\ b_8(\mathbf{C}) \end{aligned} \tag{1}$$

where f_x and f_y represent the projection of a given laser spot, whose three-dimensional location is represented by coordinates (x, y, z). The values b_1 to b_8 are functions that depend on six view parameters C_1 to C_6 as follows:

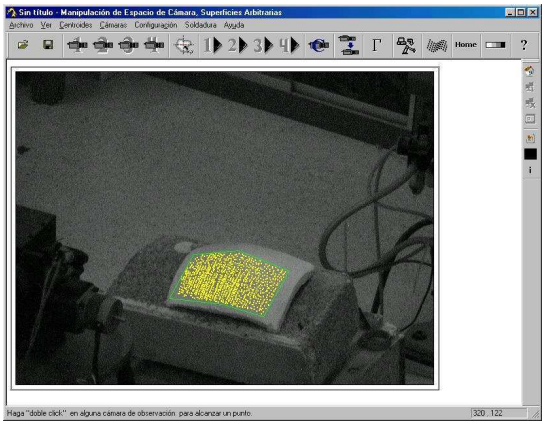

Fig. 5. User-defined polygon used to restrict the amount of information provided to the system.

$$\begin{aligned} b_1(\mathbf{C}) &= C_1^2 + C_2^2 - C_3^2 - C_4^2 \\ b_2(\mathbf{C}) &= 2(C_2C_3 + C_1C_4) \\ b_3(\mathbf{C}) &= 2(C_2C_4 - C_1C_3) \\ b_4(\mathbf{C}) &= C_5 \\ b_5(\mathbf{C}) &= 2(C_2C_3 - C_1C_4) \\ b_6(\mathbf{C}) &= C_1^2 - C_2^2 + C_3^2 - C_4^2 \\ b_7(\mathbf{C}) &= 2(C_3C_4 + C_1C_2) \\ b_8(\mathbf{C}) &= C_6 \end{aligned} \tag{2}$$

The process for estimating the view parameters is presented in detail in [6]. The estimation of the physical location of the laser spots projected over the surface is performed by considering the view parameters associated to each control camera. The estimation process is lineal and consists on minimizing the following function

$$\begin{aligned} \phi_i &= \sum_{j=1}^{n_c} \left\{ \left[x_{c_i}^{\ j} - b_1^j x_i - b_2^j y_i - b_3^j z_i - b_4^j \right]^2 \right. \\ &\quad \left. + \left[y_{c_i}^{\ j} - b_5^j x_i - b_6^j y_i - b_7^j z_i - b_8^j \right]^2 \right\} \end{aligned} \tag{3}$$

for the location (x_i, y_i, z_i) of the i^{th} laser spot. In this equation, n_c cameras detect the same laser spot, whose coordinates in the j^{th} control camera are given by $(x_{c_i}^{\ j}, y_{c_i}^{\ j})$. The values of b_1^j to b_8^j for the control cameras are presented in (2).

IV. Path Tracking

When the process of surface characterization is finished, the user selects, in camera-space, a starting point and a direction of reference over the surface, for the path to be followed. This selection is made using a graphical user interface such as the one depicted in Fig. 6. Such a point and direction will be coincident with those defined in the database, as shown in Fig. 2.

The path to be transferred over the surface is optimal in the sense that each straight-line segment, belonging to the CAD model of the trajectory, is traced as a geodesic line along the arbitrary surface. It can be shown [10] that if the

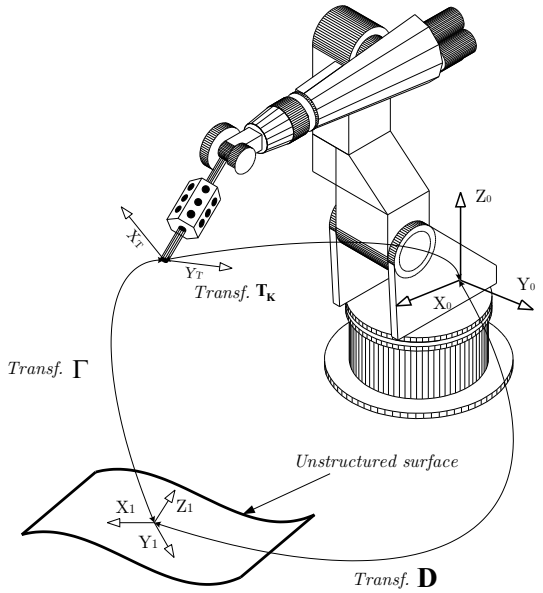

Fig. 7. Coordinate frames associated to the maneuver.

equation of the surface is expressed in the form $z = f(x, y)$, the differential equation of a geodesic line is:

$$(1+p^2+q^2)\frac{d^2y}{dx^2} = p\,t\left(\frac{dy}{dx}\right)^3 + (2ps-qt)\left(\frac{dy}{dx}\right)^2 + (pr-2qs)\frac{dy}{dx} - qr \tag{4}$$

where

$$p = \frac{\partial z}{\partial x}\ ;\ q = \frac{\partial z}{\partial y}\ ;\ r = \frac{\partial^2 z}{\partial x^2}\ ;\ s = \frac{\partial^2 z}{\partial x \partial y}\ ;\ t = \frac{\partial^2 z}{\partial y^2} \tag{5}$$

The function $z = f(x, y)$ is defined locally as a second-order, polynomial function, as follows,

$$z = a_0 + a_1 x + a_2 y + a_3 x^2 + a_4 xy + a_5 y^2 \tag{6}$$

where parameters a_0 to a_5 are estimated considering the best fit to the surface in (6), of the laser spots located inside of a restricted area, like the one shown in Fig. 5. Once this equation is determined, the differential equation (4) is solved using a numerical integration procedure like the fourth-order Runge-Kutta method, as described in [11]. The integration is performed in the direction of the i^{th} segment, until a length r_i is reached, as shown in Fig. 2. The three-dimensional location of the point at the end of the integration process constitutes the starting point of the next segment of the trajectory. The integration is performed next along a direction rotated an angle θ_i. The process is repeated for all segments included in the CAD model. The procedure for determining the robot configuration that enables an accurate following of the tool held by the robot along the geodesic lines defined above, is presented in [12]. Such a procedure involves the selection of reference frames attached to the work surface as well as to the tool held by the robot, as shown schematically in Fig. 7.

As mentioned before, a crucial requirement for the mapping of a closed trajectory, defined as a CAD model, over an arbitrary surface, is that closure must be achieved. This restriction is specially important when applied to an industrial task like cutting. The solution proposed herein for satisfying such a closure condition consists on mapping two different trajectories over the surface. One in the direction indicated by the CAD model of the closed trajectory, as depicted in Fig. 2. The second in the opposite direction. This is represented schematically in Fig. 8. With the purpose of facilitating the generation of the second path, the CAD model was modified using the following recursive formulas,

$$\begin{aligned} r'_i &= r_{n-i+1} \quad ; \quad i = 1 \ldots n \\ z'_i &= z_{n-i+1} \quad ; \quad i = 1 \ldots n \\ \theta'_1 &= \sum_{j=1}^{n} \theta_j - 180^o \\ \theta'_i &= 360^o - \theta_{n-i+2} \quad ; \quad i = 2 \ldots n \end{aligned} \tag{7}$$

where n represents the total number of segments, while θ_i r_i and z_i correspond to the i^{th} segment, as depicted in Fig. 2. The values indicated as $()'$ correspond to the second trajectory. Since the surface is assumed to be non-developable, both trajectories are not coincident. A third trajectory is generated by interpolating the vertexes corresponding to the same point of each trajectory. For instance Fig. 8 shows interpolated locations between points A and A', B and B', etc. If (X_i, Y_i, Z_i) is the 3D location of a given vertex associated to the first trajectory and (X'_i, Y'_i, Z'_i) is the corresponding vertex in the case of the second trajectory, a linear interpolation approach is proposed as follows,

$$\begin{aligned} X_{p_i} &= X_i + K_i(X'_i - X_i) \\ Y_{p_i} &= Y_i + K_i(Y'_i - Y_i) \\ Z_{p_i} &= Z_i + K_i(Z'_i - Z_i) \end{aligned} \tag{8}$$

Note that the 3D coordinates of vertexes can be obtained by minimizing (3), which will produce a location referenced to a coordinate system attached to the base of the robot. In the previous expression (X_{pi}, Y_{pi}, Z_{pi}) represents the interpolated vertex. The interpolation factor K_i is associated to the

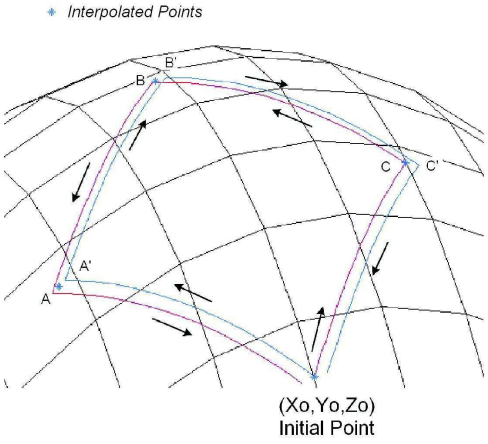

Fig. 8. Interpolation between corresponding vertexes.

i^{th} segment. This factor takes into account the accumulated error associated to the integration of the geodesic trajectories in the two opposite directions. In this paper we propose a simple approach consisting on defining the interpolation factor as,

$$K_i = \frac{i-1}{n} \tag{9}$$

where n is the number of segments of the trajectory. This factor considers basically that the interpolated point is closer to the corresponding vertex of the trajectory with a smaller accumulation of integration error. Such an accumulation of error is considered proportional to the i^{th} segment along the trajectory with a total of n segments.

Once the interpolated vertexes are obtained, the final trajectory over the surface is obtained by treating the integration of the equation of the geodesic path (4) as a boundary value problem. The integration starts at the initial point (X_0, Y_0, Z_0), depicted in Fig. 8. The shooting method [11] is applied in order to reach the first interpolated point $(X_{p_1}, Y_{p_1}, Z_{p_1})$ as close as possible. The integration continues with the next segment of the trajectory by considering as starting point of the integration process, the ending point of the previous segment. Finally, the last segment is integrated with ending point equal to the initial spot (X_0, Y_0, Z_0). This procedure ensures closure of the path, while reducing the amount of distortion of the resulting closed trajectory. Fig. 9 presents a simulation obtained using this procedure, with a simple path. This path is experimentally traced, as discussed in the following section.

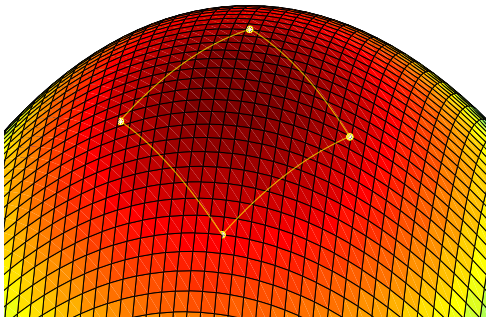

Fig. 9. Simulated path through interpolated points.

V. Experimental results

An industrial Fanuc ArcMate 100i robot was used to test the methodology proposed in this paper. This robot has a tool with known geometry, attached to a welding torch. Such a tool, depicted in Fig. 10, has manipulated features placed on it, which are used to facilitate the process of image-analysis.

The communication between the robot and the computer where the control algorithms were developed, is performed using a TCP/IP protocol. For the work presented here, the communication is required in order to get samples of the robot configuration while this is moving towards its objective. It is also required with the purpose of commanding the robot to the configuration evaluated with the method presented in the previous section.

For the surface-characterization process, a commercial, Lasiris laser diode of $5mW$ and $635nm$ is used. This device has a diffraction head that divides the laser beam into a matrix array of 7 X 7 laser beams. In order to characterize the surface and to evaluate the internal configuration of the robot, two cameras are required. When the laser spots are detected with the two Sony MPT304, CCD control cameras used in the maneuver, these are seen as white spots. The images were digitized using a Data Translation DT3155 frame-grabber, installed inside of the PC compatible computer where the control algorithms were developed.

For the experiments reported here, two testing paths were considered, both with a shape of a square. The first one with a side length of 30 mm and the second with a side length of 60 mm. These paths are shown in Fig. 11. It is important to mention that there is no restriction regarding the size or complexity of the shape. The only restriction is that the cameras must have visual contact with the surface where the path is going to be traced.

The testing trajectories were drawn over a surface made of clay, with a spherical shape of approximately 120 mm of external radius. As shown in Fig. 12, the results obtained are a good indication of the effectiveness of the proposed methodology. The results were satisfactory as the error between the traced trajectory and the one stored in the database is in the order of ± 0.5 mm.

In order to compare the effectiveness of the proposed methodology, two closed trajectories were drawn over the same surface. The first closed trajectory was performed without applying the measure for achieving closure, described here. The second one was traced with the proposed technique. Fig. 13 shows the resulting paths, for each of the two testing trajectories. As can be seen from this figure, when a geodesic mapping is applied without the measure proposed here, the separation between the initial and final point of the

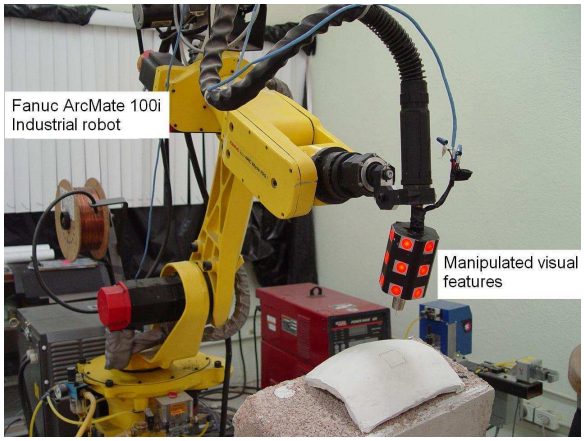

Fig. 10. Industrial robot used in experiments.

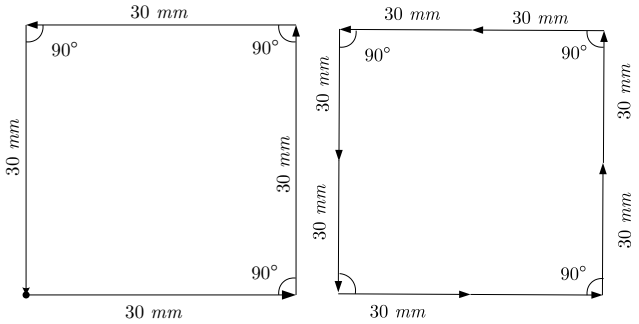

Fig. 11. Testing trajectories.

closed trajectory is about 2.5 mm, in the case of the square with a side length of 30 mm. This separation doubles when the square of 60 mm is traced. Although an apparent linear relationship between these two results exists, no conclusions can be drawn until further experiments are performed with a larger and more complex shape. Again, the proposed methodology is not restricted by the size or the complexity of the path. It should be noted that if the path traced without the methodology proposed herein was associated to a cutting task, the result would have been completely unsatisfactory. When the measure is applied, closure is accomplished.

VI. Conclusions

The experiments performed to date enable us to establish that the proposed measure is a viable alternative to project an arbitrary closed trajectory over a curved surface of unknown geometry. Such a projection is made in an optimal fashion, using the concept of geodesic mapping combined with a weighted interpolation between two opposite trajectories.

The experiments also enabled the possibility of programming an industrial manipulator with a device other than a teach pendant. In this case, the programming of the robot is facilitated with the method presented here, and this program can be repeated only once or multiple times. This fact contrasts with the applications given to most industrial robots where a single task is repeated in multiple occasions. Since the designation of the task can be performed in an asynchronous fashion with respect to the actual development of the industrial maneuver, then the remote programming of the system via internet is a viable alternative.

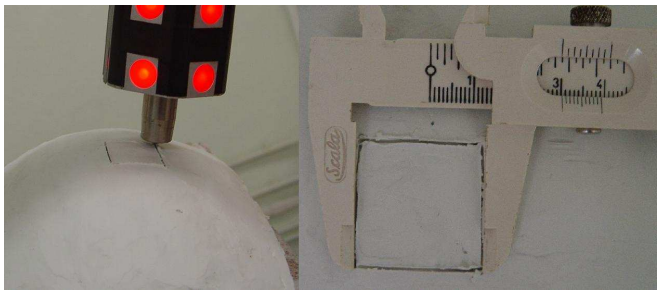

Fig. 12. Smaller testing path traced over a soft surface.

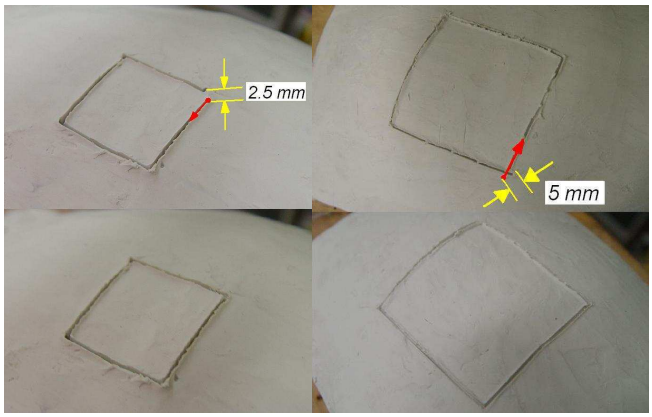

Fig. 13. Comparison between closed paths traced without the procedure proposed herein (top) and with the proposed technique (bottom). Left column corresponds to the 30 mm square, right column to the 60 mm square.

Finally, in the experiments presented here, once an initial set of robot configurations that enable a closed path is defined, it is expected that an industrial task would follow. The development of such a task, like cutting, can proceed by defining in advance adequate process parameters like traveling speed, voltage, etc. This in turn will enable that a user, without special qualifications in robotics or in the particular industrial task, will be able to successfully program the system.

References

[1] M. Seelinger, E. Gonzalez-Galvan, M. Robinson, S.B. Skaar, “Towards a Robotic Plasma Spraying Operation Using Vision”. IEEE Robotics and Automation (Special Issue on Visual Servoing), 5(4), 1998, pp. 33-36.

[2] Cisneros F. Engineering Manager. Yaskawa-Motoman Mexico. Personal communication. 2003.

[3] E.J. González-Galván, A. Loredo-Flores, F. Pazos-Flores, J. Cervantes-Sánchez, “An Optimal Path-Traking Algorithm for Unstructured Enviroment based on Uncalibrated Vision”, IEEE International Conference on Robotics and Automation, April 2005, pp. 2547-2552.

[4] H.W. Guggenheimer. Differential geometry. Dover, New York, 1995.

[5] http://www.fanucrobotics.com/file_repository/fanucmain/adapt_weld.pdf

[6] E.J. Gonzalez-Galvan, S.B. Skaar, “Efficient Camera-Space Manipulation Using Moments”. IEEE International Conference in Robotics and Automation, 1996, pp. 3407 - 3412.

[7] M. Seul, L. O’Gorman, M.J. Sammon. Practical Algorithms for Image Analysis. Description, Examples and Code. Cambridge University Press, 2000, pp. 86-89.

[8] J.R. Parker. Algorithms for Image Processing and Computer Vision. John Wiley & Sons, 1997.

[9] D. Stücker, “Elementary geometric methods: Line Segment Intersection and Inclusion in a Polygon”. Technical Report. Department of Computer Science. University of Oldenburg. 1999.

[10] I.N. Bronshtein, K.A. Semendiaev, Handbook of Mathematics. 3rd Edition. Van Nostrand Reinhold Co. 1985.

[11] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes. The Art of Scientific Computing. 2nd Edition. Cambridge University Press. 1992.

[12] E.J. Gonzalez-Galvan, S.B. Skaar, M.J. Seelinger, “Efficient Camera-Space Target Disposition in a Matrix of Moments Structure Using Camera-Space Manipulation”. The International Journal of Robotics Research. Vol. 18, No. 8, August 1999, pp. 809-818.