'87 KACC 1987.10.16~17

Two-Dimensional Object Contour Tracking by a Force Controlled Manipulator

°Myoung Hwan Choi , Myoung Sam Ko and Bum Hee Lee

Robotics & Intelligent Systems Lab.
Dept. of Control and Instrumentation Engineering
Seoul National University, Seoul, Korea

Abstract

The ability of a robotic manipulator to recognize the shape of an object by feeling its hand around the object is useful in many applications. Two-dimensional object contour tracking by force feedback is described. The system consists of IBM PC/AT, PUMA 560 manipulator, PUMA controller and a tip sensor. Position control is accomplished by using VAL command and the unmodified PUMA controller. A contour tracking algorithm is developed and tested on three different types of objects. The experimental results show that the objects' shapes can be successfully identified .

1. Introduction

In today's industry, robots are playing increasingly important role in manufacturing and automation. The application of the robots, however, are mostly limited to the simple repetitive tasks that can be taught by pre-programing, and requires little or no intelligence. Efforts are being made by many researchers to increase the intelligence of robots and expand their application areas.

One way of giving intelligence to a robot is to enable it to identify the shape of an object by feeling its hand around the object. The problem of tracing the edges of an object is related with force control and generation of manipulator paths based on the sensor information. The edge following has been studied elsewhere, notably by Stepien[1], Wampler[2] and Starr[3]. The work by Stepien treated the deburring problem, and that of Wampler was a two dimensional object contour follower using an optical proximity sensor. Both works involved modifications to the robot controller, while the work by Starr used VAL-II and unmodified robot controller.

In this paper, two dimensional object contour tracking by force feedback is described, where a manipulator tracks the edges of a planar object, and identifies its shape. A tip sensor was constructed to detect contact force in X-Y plane, and mounted at the end of the manipulator. PUMA 560 manipulator was used for experiment, and IBM PC/AT was connected to PUMA controller and it processed the sensor signal and generated tool position command. Position data along the contour of the object were recorded and later plotted by X- Y plotter.

2. System Setup.

System block diagram is shown in Figure 1. A tip sensor produces two channels of force information(Fx and Fy), which are amplified by a strain amplifier. Amplified force informations are converted to digital data and processed by IBM PC/AT. IBM PC/AT is connected to the CRT terminal line of the PUMA controller through a serial communication port, and sends out VAL's immediate move commands. IBM PC/AT carries out necessary calculations for force information processing, contour tracking algorithm and communication with the PUMA controller. PUMA controller is in charge of manipulator position control. Softwares are written in assembly and C language.

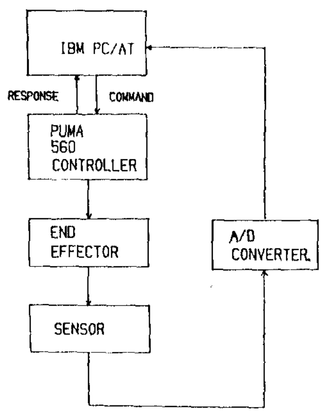

Figure 1. System block diagram

3. Force control algorithm

Much work has been done by many researchers on the subject of force control for robotic manipulators[4]. Force control scheme that was utilized in this study is the one that uses a combination of joint position sensors and a tip mounted force sensor. Systems of this category can be classified[5] into joint based or cartesian based systems by the coordinate system in which the control algorithm is based, or alternatively into torque based, velocity based or position based systems by the method in which the force sensor signal is processed through the force servo loop.

A control scheme, in general, is said to be cartesian based if the error between desired and actual manipulator positions is formed in cartesian space . Conversely, joint based systems form this error in joint space. In torque based systems, the signal from the force sensor is processed directly to become a torque command. In velocity based systems, velocity command is computed from the force sensor signal, and position command is computed from the force sensor signal in position based system.

Among various control schemes described above, the scheme suggested by Maples and Becker[5] was adopted for force control in this study for following reasons : First, cartesian based scheme was chosen as it was more natural way for force control as opposed to the joint based one. Second, position based scheme was prefered for reasons of good disturbance rejection and ease of implementation under IBM PC/AT - PUMA controller - sensor configuration which was the basis of the experiment in this study.

The force control algorithm is outlined in Figure 2. The algorithm uses PUMA position servos for position control and force control.

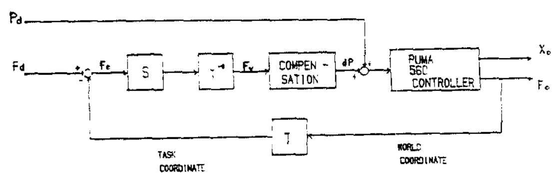

Figure 2. Force servo loop.

P_d = desired position

F_d = desired force in task coordinate frame

dP = path correction command

S = selection matrix

In Figure 2, commands are given for position and force. Force signal in the world coordinate frame is transformed to the task coordinate frame. Transformed force is subtracted from force command, and force error is multiplied by a selection matrix,S,[6], to separate the axis being force controlled from the axis being position controlled. Then the error is transformed back to world coordinate frame, and processed to become a path correction to feed to the position controller.

The compensator used was a simple accumulator with a gain, $K \cdot \frac{z}{z - 1.0}$. The force error determines the velocity of the probe.

Task coordinate frame is defined as in Figure 3. Origin is the contact point, and X_t - axis lies along the direction of contact force. Z_t - axis lies along the upward vertical direction which is parallel to the world Z - axis of PUMA 560 manipulator. Y_t axis is defined by the right hand rule and lies along the surface tangent. As the contact point moves along the object edges, the task coordinate frame rotates. Force control is applied along X_t - axis and position control is applied along Y_t and Z_t axis. The orientation of the probe itself is maintained constant throughout the tracking.

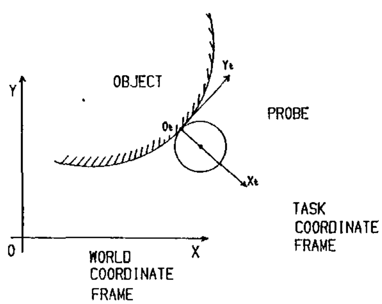

Figure 3. Task coordinate frame

4. Contour tracking algorithm.

The basic goal of the contour tracking algorithm is to have the robot feel its way around an irregularly shaped object to determine its contour. Once the contour information is obtained, it can be stored for viewing or further processing at later time.

Following assumptions are made: First, objects are planar in shape, i.e. objects are confined to two dimensional objects. Second, the contacting tool(or probe) approaches the object for contact, from the outside of the object, i.e. from the outside of its convex hull. Third, contact force between the contacting probe and object surface is normal to the surface. Fourth, Objects are assumed to have simple shapes. Fifth, at the beginning of the tracking, the manipulator moves along the pre-set approach direction, and the object is assumed to be in that direction.

Under the above assumtion, the contour tracking algorithm works as in Figure 4. The manipulator moves to a standby position which lies on the plane of the object, The manipulator is now ready to approach the object. The force error between the commanded contact force and the actual force drives the manipulator towards the object. Contact occurs between the probe and the object, and the contact force is servoed to the contact force command Fd. Contact position is recorded when the contact force F_o enters the force range F_d - dF1 < F_o < F_d + dF2.

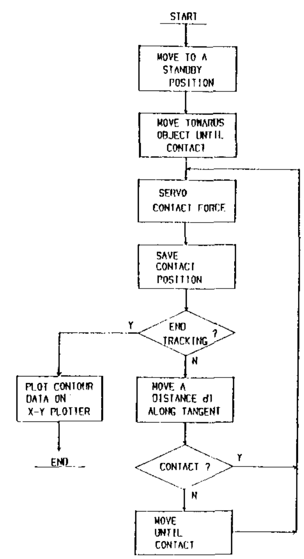

Figure 4. Contour tracking algorithm

After the position data storage, tracking termination condition is tested . The termination condition is shown in Figure 5 a). The basic concept is to set a reference point O inside the object in the probe's approach direction after the first contact, and terminate the tracking when the contact point encircles the point O. The distance OXo is equal to the radius of the probe. The angle through which the probe has moved is monitored and the tracking stops when the angle is greater than or equal to 360 degrees. The validity of this termination criteria can be shown from the second and fourth assumption.

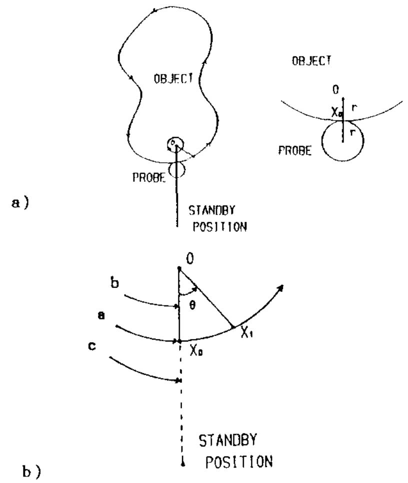

Figure 5. Tracking termination

If the contouring starts at point Xo and the reference point is O, there are three possible ways for the probe to encircle the reference point[Figure 5 b)]. In path a, the probe encircles the reference point and successfully completes the tracking. In paths b and c, the probe encircles the point O, but does not meet the starting point Xo, and ends the tracking prematurely. However, the path b is disregarded by the fourth assumption. Path c cannot occur, because of the second assumption, in other words, there can not be an object surface between the standby position and the first contact point Xo. If the tracking is terminated, the contour of the object is plotted on a X-Y plotter. Otherwise, the probe is moved a small distance dl along the surface tangent, and contact force is examined. If the probe is still in contact with the object, the force servoing begins and the loop goes on. If the probe looses contact, the probe searches the object in the following way[Figure 6].
Let P0 be the previous contact point with the object, and P1 be the present point away from

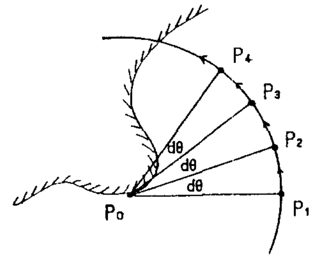

Figure 6. Re-contact strategy.

the object. Then the probe moves along the circle of radius dl, origin P0, through an angle dθ at a time, i.e. it moves through points P2, P3, ... until contact is made with the object.

5. Tip sensor

A tip sensor was constructed for the tracking purpose. the Tip sensor detects force in X-Y plane, and produces two channels of signal(Fx, Fy). The sensor[Figure 7] can be divided into three parts: mount, strain gauge area and contacting area. The gripper of the PUMA 560 manipulator was removed and the tip sensor was mounted in its place. Contacting area is where the actual contact is made with the object, and is cylindrical in shape. Two pairs of strain gauges convert mechanical deformation into electrical signals. Strain amplifier amplifies the strain gauge signal and feed it to A/D converter. IBM PC/AT reads the digitized force signal and use it for contour tracking.

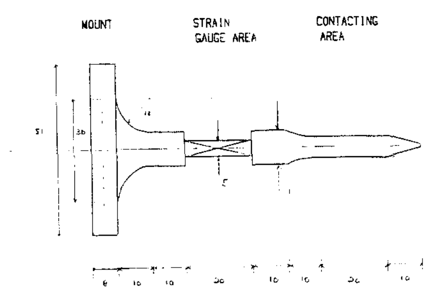

Figure 7. Tip sensor

6. Bxperiments and Results

Experimental setup is shown in Figure 8. PUMA 560 manipulator is used and the tip sensor is mounted in the place of the gripper. Object is fixed on a table, and the probe of the tip sensor traces around the object. The algorithm that was developed to trace the contour of the object worked quite well. The probe followed around the object and never became lost. The process of tracing the entire object was slow due to the slow communication between the computer and the PUMA controller, and the time taken to execute the move command which was about 0.4 second. This limitation is due to the dependence on VAL, and can not be overcome unless the controller is modified.

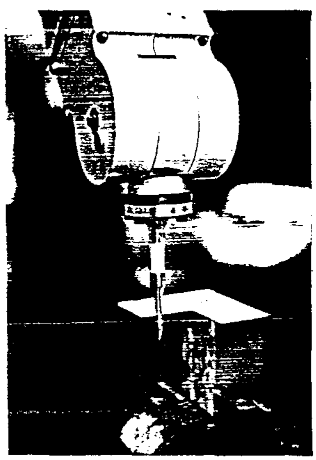

Figure 8. Experimental setup.

Typical parameters used are as follows.

incremental distance, dl = 1.0 mm

incremental angle, $d\theta$ = 15 degree contact

force command, F_d = 0.39 N

contact force range : 0.19 N < F_o < 0.59 N for position storage

compensation gain , K = 0.008

Three different objects were traced, and their contour as viewed by the manipulator were plotted on a X-Y plotter. The plotted results are shown in Figure 9.

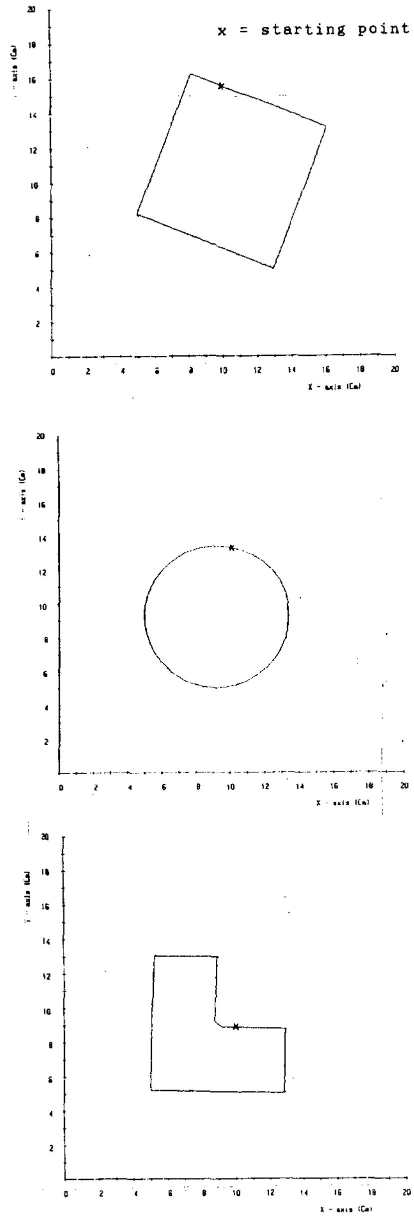

Figure 9. Object contour

Inspection of the plot shows four types of noticeable errors. First, object edges are shown as jagged lines. This erorr can be attributed to the violation of the third assumption, i.e. friction exists between the contacting probe and object edge, and the contact force is not confined to the surface normal direction . Second, object edges at the sharp convex corners show scattered contact points and overlapping lines. This can be caused by the fact that object compliance increases at the sharp corners, and the direction of contact force at the sharp corner points also violates the third assumption. The details of sharp extruding corners can also be lost due to the finite incremental distance. Third, sharp concave corners are cut away[Figure 9 c)]. This error is due to the radius of the probe. Smaller radius reduces this type of error, but the practical difficulty of its construction increases, and compliance of the probe also increases which leads to positional errors.

7. Conclusions.

Two - dimensional object contour tracking by force feedback was implemented with IBM PC/AT, PUMA 560 manipulator, PUMA controller, and a tip sensor. Contour tracking algorithm was developed and tested on three types of objects, and successful results were obtained. Two-dimensional object contour tracking lays the base for three-dimensional object contour tracking. Object identification by contour tracking will provide an alternative means to visual methods, which are plagued by problems such as lighting, hidden surfaces, depths etc.

Two-dimensional object contour tracking can be extended to find the shape of the three-dimensional object. One of the ways is to find a series of cross sectional contour of the object and deduce the contour of the three dimensional object. Research along this line and further analysis is under way

8. References

[1] T.M. Stepien, L.M. Sweet, M.C. Good, M.Tomizuka, "Control of Tool/Workpiece Contact Force with Application to Robotic Robotic Deburring," Proc. 1985 IEEE Int. Conf. on Robotics and Automation

[2] C. Wampler, "Multiprocessor Control of a Telemanipulator with Optical Proximity Sensors," Int. J. Robotics Research, Vol 3, No.1, Spring 1984

[3] G.P. Starr, "Edge Following with a PUMA 560 Manipulator Using VAL-II," Proc. 1986 IEEE Int. Conf. on Robotics and

[4] D.E. Whitney, "Historical Perspective and State of the Art in Robot Force Control ," Proc. 1985 IEEE Int. Conf. on Robotics and Automation

[5] J.A. Maples, J.J. Becker, "Experiments In Force Control Of Robotic Manipulator, " Proc. 1986 IEEE Int. Conf. on Robotics and Automation.

[6] M.H. Raibert, J.J. Craig, "Hybrid Position/Force Control Of Manipulators, " Tran. ASME J. Dyn. Syst. Measurement and Control, Vol 102, June 1981