

# A 2D unknown contour recovery method immune to system non-linear effects

Sergi Hernandez Juan  
Universitat Politcnica de Catalunya  
Institut de Robtica i Informtica Industrial (IRI)  
Barcelona, Spain  
Email: shernand@iri.upc.edu

Josep Maria Mirats Tur  
Universitat Politcnica de Catalunya  
Institut de Robtica i Informtica Industrial (IRI)  
Barcelona, Spain  
Email: jmirats@iri.upc.edu

**Abstract**—A method to recover general 2D a priori unknown contours using a kind of special optic sensor is described. Contour recovery is an important task for exploratory operations in unknown environments as well as for more practical applications such as grinding or deburring. It is not an easy task since the recovered contour (generally obtained using encoder data) is severely distorted due to errors in the kinematic model of the robot and to the non-linearities of its actuators. Some mathematical models have been presented to partially compensate for those effects, but they require a deep knowledge of both the robot and sensor models which are difficult to obtain accurately, and normally imply an adaptive non-linear control to estimate some of the unknown parameters of the model.

Our approach, in despite of its simplicity, is intrinsically immune to non-linearities, which allows us to eliminate most of the distortions added to the sensor data. A simple algorithm to follow unknown planar contours is presented and used to test the performance of this approach in comparison to the one using encoder data. Experimental results and practical problems are also discussed.

**Index Terms**—Contour recovery, PSD sensor, Robotic arm manipulator.

## I. INTRODUCTION

Contrary to humans, robotic manipulators do not have good abilities in dealing with unknown shape objects. Robot manipulators used in industrial facilities assist or even substitute people in tedious and repetitive tasks in structured environments, usually working as programmable devices that reproduce previously recorded motions. However, there exist a number of complex industrial applications in unstructured environments such as grinding [1], deburring [2], polishing and automated manufacturing inspection [3] which may need to obtain an unknown object contour and, sometimes, follow it with an specific applied normal force.

In a general case, there are basically two methods for obtaining the shape of an unknown object, the first one is using computer vision [5]; and the second is using a robot to follow the contour using some kind of tactile sensor and odometry information [6] and [7]. Combined methods using both vision and force information have also been developed [8].

The former method embeds a great variety of techniques such as structured light [9], binocular stereo, shading [10] and photometry [11] just to name some. All these techniques

exploits the information contained in the raw image data to obtain the shape of the object. These kind of methods are very sensitive to environmental conditions and also to the contour's material properties (reflectness for example). These techniques also are very expensive computationally, implying, in practice, speed limitations. It is beyond the scope of the present work to discuss further such methods.

When a tactile sensor is used, the shape of an unknown object can be detected by directly touching the object. Depending on the kind of tactile sensor used, different information about the shape of the touched object can be obtained. If a force/moment sensor is used, usually based on strain gauge devices [6], we can obtain data about location and different force components on the surface, depending on the number of axis where forces are being measured. Array tactile sensors are also available, which give local shape from a single contact measurement [12].

Precise unknown shape reconstruction is still an open problem, even though a number of works discussing this issue may be found using fingertip or array sensors. For instance, in [13] a method using a network of triangular B-spline patches based on arbitrary topological domain maintaining tangent plane continuity is used. Techniques for recognizing convex polyhedra from sparse tactile data, using volumetric representations, are used in [14]. The performance of an hybrid force/velocity controller for automatic edge following is analyzed in [15]. In [16] a method of recognizing geometric shape of a polyhedron has been proposed using position and surface normals of multiple measured contact points. The use of array sensors is reported in [17] where a method of discriminating between contact with planar surface and a sharp edge is developed.

For industrial applications fingertip force/moment sensors are preferred for their convenient price and easiness of use. As a fingertip sensor provides local information, it is necessary to keep a continuous touching (sliding or rolling) on the object while moving the fingertip through it. Several problems arise here such as slipping between the fingertip and the object, different friction constants for different object materials, or robot non desired mechanical effects as backlash or dead-zones. Some mathematical models are introduced in [3], [4] and [18] in order to partially compensate some of this

effects, but they lead to computationally expensive algorithms, normally suited only for specific shape properties, robot architectures and sensor models.

In this paper we present a novel method to obtain the contour of an unknown object based on combining a simple force surface following algorithm and an external optical Position Sensing Device (PSD). By using a kind of visual feedback, the method is intrinsically immune to most non-desired effects due to robot architecture and non-linearities and force sensor model, but still keeping the computational complexity low.

This paper is organized as follows: section II describes the mark of the project where this research has been taken place as well as the used platform and sensors; section III is devoted to the contour following algorithm developed; section IV shows experiments and results and finally section V concludes the paper.

## II. SYSTEM DESCRIPTION

The robot platform used to implement and test the contour recovery approach presented in this paper consists of a 3 degrees of freedom (dof) modified SCARA architecture robot. As shown in Fig. 1(a), the designed manipulator has four joints, with the particularity that two of them have been attached together (joints labeled  $O$  and  $O'$  in Fig. 1(a)) so the movement of point  $T$  is lineal along axis  $x$ . The other two degrees of freedom are used to control the pan (joint  $T$ ) and tilt (joint  $Q$ ) of the end effector. The physical implementation is shown in Fig. 1(b)

Three actuators are needed to move the robot articulations. Chosen devices have been DC motors with a planetary gear box including an incremental magnetic encoder in the motor shaft in order to feedback the axis position.

Additionally, a 2 dof force sensor based on strain gauges has been fixed to the end effector of the manipulator. It allows measuring the force components that are transversal to the robot end effector. The whole system is completed with an external optical position feedback sensor based on a 2D Position Sensing Device (PSD) described in detail in section II-A.

The use of gears in each joint causes backlash to appear. This effect adds a certain level of compliance to the end effector in the direction of the motion, making the object contact safer, but, on the other hand, adding a considerable uncertainty in the robot's end effector position, and thus, making virtually impossible to obtain the real contour from encoder data. The typical backlash model is presented in (1) and Fig. 2.

$$\dot{\tau} = B(\tau, u, \dot{u}) = \begin{cases} m\dot{u}, & \text{if } \dot{u} > 0 \text{ i } \tau = mu - md_+ \\ & \text{if } \dot{u} < 0 \text{ i } \tau = mu - md_- \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Because of the limitations in the number of degrees of freedom of both the robot and the force sensor, all the

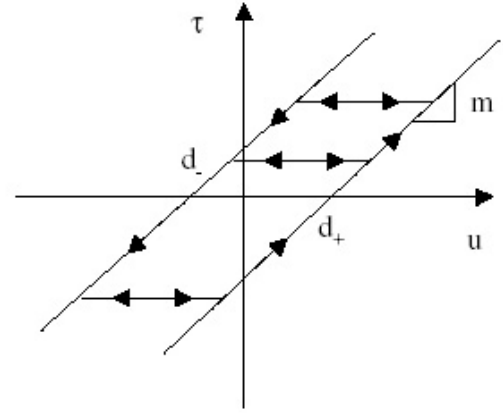


Fig. 2. Backlash input/output model

experiments presented later in section IV are restricted to a 2D space but all the concepts can be easily generalized to a 3D case. This fact forces all the robot movements to be restricted into an  $YZ$  plane maintaining a constant distance to the robot's framework origin.

### A. Optical Position Sensing Device

The used Position Sensing Device (PSD) internal structure (shown in Fig. 3) is similar to that of a PIN photoreceptor, that is, a layer made of semiconductor type P as the anode, an intermediate layer made of intrinsic semiconductor to boost the light-current conversion efficiency, and finally, a semiconductor type N layer as the cathode. The main differences between this kind of sensor and a traditional photoreceptor are its active area (from few  $mm^2$  to some tens of  $mm^2$ ), its shape (normally round or square), and the number of terminals (4 cathodes and 1 common anode).

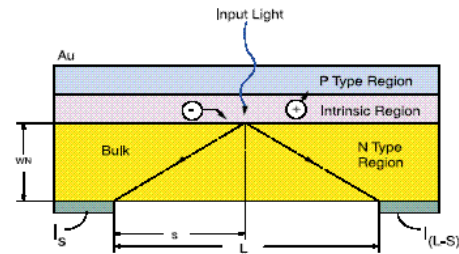
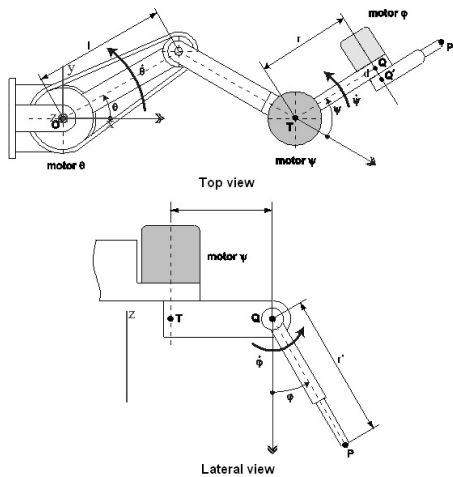
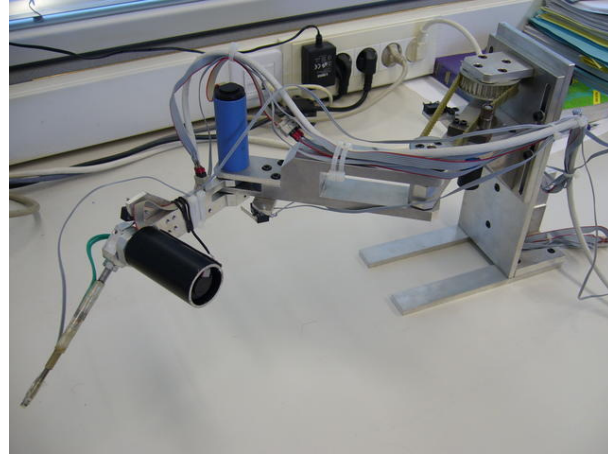


Fig. 3. Internal structure of a Position Sensing Device (PSD).

The working principle of such devices is that the photo-generated current in each terminal is directly proportional to both, the position and the intensity of the incident point of light on the sensor's active area. Measuring the output current for each of the sensor terminals and using (2) and (3), it is possible to obtain the precise position  $(u, v)$  of an incident spot of light. These expressions relates the  $(u, v)$  position to  $L$ , the sensor length, and each one of the pair currents  $((I_1, I_2)$  and  $(I_3, I_4))$  measured in opposite sides of the sensor.



(a) Top and lateral view of the robot architecture.



(b) Physical implementation of the robot architecture.

Fig. 1. Robotic platform used in this paper.

$$u = \frac{L}{2} \frac{I_1 - I_2}{I_1 + I_2}. \quad (2)$$

$$v = \frac{L}{2} \frac{I_3 - I_4}{I_3 + I_4}. \quad (3)$$

As it usually happens with all the existing devices, the PSD has advantages and disadvantages. A positive aspect is, for instance, the elevated update rate that can be achieved, up to one mega-samples per second. The refresh ratio is only limited by the conditioning circuitry and the required signal post-processing calculations, just a few arithmetic operations for this kind of devices. This kind of sensors also have a certain immunity to the desired light source intensity because of the use of relative information in (2) and (3). Static as well as dynamic undesired light sources in the work space can be easily compensated by modulation of the used light source and a sensor calibration stage.

One of the biggest drawbacks associated to the use of these kind of sensors is the necessity of modifying the work zone by placing light sources in the points whose coordinates are to be retrieved. As the sensor gets the coordinates of the centroid of the incident light, only one punctual light source can be active at a time. This implies the serialization of the position feedback procedure and also the need to control the state of the desired light sources to turn them on and off.

An other important drawback is the extremely sensitivity needed in the conditioning circuit to detect and amplify the sensor's output current levels, making the system also highly sensitive to noise and random interferences. Also, even with its selective sensibility, the PSD generates valid readings for all the visible and near IR spectrum, which forces the use of optical filters. Finally, it also needs a lens to focus on

the work zone, which has to be integrated with the filter, the conditioning circuitry and the sensor itself in order to use it.

In the present work, in order to be able to recover an unknown contour with PSD sensors, a punctual light source (actually, a 670 nm Light Emitting Diode (LED) ) has been placed at the edge of the robot's end effector, and the robot has been ordered to follow the unknown contour.

Once assembled, the PSD camera must be calibrated to find the extrinsic and intrinsic parameters, that is the translations and rotations with respect to the robot's framework and the focal length and the non-linear distortion parameters respectively. There exist a number of calibration techniques for CCD cameras ([19], [20] or [21]), but none of them can be used because they don't accurately model the non-linear distortion exhibited by the PSD sensor.

Given the limitations imposed in the work zone of the robot, it's possible to use only one PSD camera, placed parallel to the  $YZ$  plane where the robot's motion is limited, in order to get the followed contour. Thus the calibration process needs only to provide two of the space coordinates ( $y$  and  $z$ ), since the  $x$  coordinate information is implicit in the calibration process itself.

So, a standard non-linear Least Square technique has been used to minimize the fitting error of a function in terms of the sensor coordinates  $(u, v)$  ( $f_z(u, v)$ ,  $f_y(u, v)$ ), given a set of points in both the sensor and the robot's framework coordinate systems. The functions used to translate the coordinates between the two coordinate systems are complete 3 degree polynomial on  $u$  and  $v$  in order to compensate for sensor non-linear distortions. Once the calibration is completed, all the least significant terms are eliminated to reduce computational requirements.

### III. SURFACE FOLLOWING ALGORITHM

Typically, two main phases can be recognized in a complete edge-following task. In a first step, contact with the environment is sought; to this purpose, the end effector is moved towards the object of interest and a safe impact is attempted. In order to preserve the contact, a suitable steady value for the interaction force is assigned. The second step is aimed to maintain a desired value for the contact force while the end effector moves along the environment surface; during this phase, the motion of the end effector results from the accommodation of the planned motion to the mechanical constraints imposed by the environment.

The design of a single algorithm capable of following any 2D contour only using force information is quite complex since there is a lot of possible cases. Using the platform presented in section II most of these cases, due to hardware limitations, are extremely difficult or even impossible to detect, so a more simple algorithm is presented which is capable of following contours with a positive or negative slope of up to 90° but not higher.

The motion commands to the robot are velocities ( $\dot{y}$  and  $\dot{z}$ ) instead of positions to smooth the robot's trajectory, and the inputs to the control law are both measured forces ( $F_y$  and  $F_z$ ). The default expressions to get both the  $y$  and  $z$  linear speeds are:

$$\begin{cases} \dot{z} = K1_z \cdot (F_z - Th1_{Fz}) \\ \dot{y} = Speed_y \end{cases} \quad (4)$$

In the above expression,  $Th1_{Fz}$  is the normal desired force in Newtons in the  $z$  axis. Any value  $F_z$  read from the force sensor different from that will force the robot to move in that direction to reduce the gap. The way this compensation is carried out depends on the value of  $K1_z$ . A large value on this variable could make the robot's end effector to oscillate, and a small value could make the robot to lose contact with the surface.

Equation (4) are used while  $F_y$  is small (i.e. it is the result of noise or friction between the end effector and the contact surface) and  $F_z$  is close to the desired contact force.

When  $F_y$  rises above a certain threshold (i.e. a positive slope has been reached) two new terms are added which depend on that force. These new terms, as shown in (5), slow down the motion in the  $y$  axis to avoid a sudden increase in  $F_y$  (the term  $K2_z \cdot F_y$ ) while compensating the desired contact force in the  $z$  axis to allow the robot to rapidly gain height in case of a heavy slope (the term  $K_y \cdot F_y$ ).

$$\begin{cases} \dot{z} = K1_z \cdot (F_z - Th1_{Fz}) + K2_z \cdot F_y \\ \dot{y} = Speed_y + K_y \cdot F_y \end{cases} \quad (5)$$

In case of a negative slope, and due to the constant force applied to the surface,  $F_y$  may fall below a certain threshold. In this case, to avoid losing contact with the surface, the motion in the  $y$  axis is purely proportional to  $F_y$ , which allows the robot to stop whenever it loses contact with the surface as shown in (6).



(a) Sinusoidal contour

(b) 90° slope contour

Fig. 4. Contours used to test the PSD based contour recovery approach.

$$\begin{cases} \dot{z} = K1_z \cdot (F_z - Th1_{Fz}) \\ \dot{y} = K_y \cdot F_y \end{cases} \quad (6)$$

Two special cases are found when  $F_y$  rapidly increase (about 90° positive slope) and when  $F_z$  falls close to 0 (about 90° negative slope). In those cases the contour following algorithm is the same just presented, but switching the  $y-z$  axis. That is, the steady force value is applied on the  $y$  axis, and the motion is along the  $z$  axis.

### IV. EXPERIMENTS AND RESULTS

The approach presented in this paper has been tested using two different kinds of contours shown in Fig. 4: a sinusoidal like contour (Fig. 4(a)) and a 90° slope contour (Fig. 4(b)). Both contours are intended to test the contour recovery capabilities of the encoder data based traditional method and the new one using PSD data. Non desired non-linear effects in the recovered contour when using encoder data are shown. When using the PSD based approach these effects are eliminated, but new problems arise. In both experiments the robot's end effector follows a trajectory in the  $YZ$  plane, from  $y = 100 \text{ mm}$  to  $y = -100 \text{ mm}$  modifying the  $z$  coordinate according to the contour.

We show in Fig. 5 the sinusoidal contour reconstructed from the encoder data (in blue) together with the contour obtained from the PSD data (in green) and the real contour (in red). It's clear that the reconstruction from encoder data is more accurate in terms of shape, but the position of the contour in the robot's framework is displaced about 25mm to the left. This effect is mainly caused by the delay in motion between the joint's actuator and the joint itself due to mechanical backlash.

On the other hand, the contour reconstruction using PSD data is noisy, but the absolute position of the contour in the robot's framework is more accurate. This is because while the joints are inside the dead zone produced by the backlash, no actual movement is achieved and none is detected by the external optic feedback system (PSD), but the encoders do detect a false displacement since they measure the rotation of the actuator axis.

It is important to notice that data obtained with the optical sensors degrade as the light source is moved out of the central region of its active area. This is mainly because the source

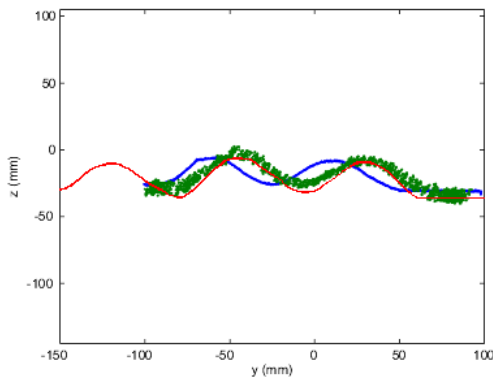


Fig. 5. Sinusoidal contour reconstruction using encoder data (blue) and PSD data (green). The real contour is plotted in red.

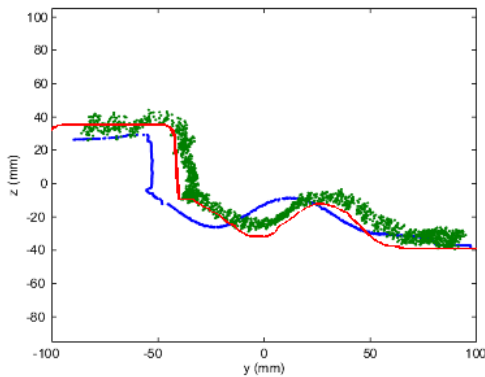


Fig. 6.  $90^\circ$  slope contour reconstruction using encoder data (blue) and PSD data (green). The real contour is plotted in red.

itself changes its heading from the PSD camera framework as the robot moves, and the light source radiation diagram is far from an ideal omnidirectional source.

As happens with CCD cameras, the PSD feedback system is sensitive to calibration errors, but contrary to the former methods, they are quite immune to environmental changes like illumination. As for the small signal to noise ratio, it may be easily improved by using a better conditioning circuit, which are commercially available, and also by adding some post process stage to the incoming data to filter some of the noise out.

Similar results can be obtained in following a  $90^\circ$  slope contour as shown in Fig. 6. That figure shows its reconstruction using encoder (blue) and PSD data (green). The real contour is also plotted in red.

A part from the same obvious spacial displacement between the two contours, here it's important to notice that whenever the robot abruptly changes its moving direction, additional undesired non-linear effects appear (mainly due to mechanical backlash). As it can be appreciated in Fig. 6, there is an overshoot in each  $90^\circ$  corner that are not present in the PSD

contour reconstruction.

## V. CONCLUSIONS

The simple force based contour following algorithm presented in this paper successfully followed all the tested contours and allowed the authors to verify all the advantages and drawbacks of this new approach.

Although the data obtained from the PSD sensors is quite promising, it's highly sensitive to the conditioning circuitry used, which requires precise adjustments. We have learned that, for a surface following task, although encoder data is generally less noisy, the PSD data is best suited to obtain the real position and, sometimes, the shape of the contour in the robot's workspace. However, neither the encoder nor the PSD data based approaches are good enough by themselves to ensure an error free reconstruction of the really followed contour.

A good example of a practical application where such techniques are needed is deburring a work piece in order to obtain an object with a determined form, for instance, obtaining a ship hull from a piece of raw carbon fiber material. These tasks are usually done in a hierarchical form, that is, the deburring tool removes the material in an incremental way passing through the surface different times until it reaches the desired form of the object. In order to perform this process robustly, a method that is capable to obtain the generated contour in each step is needed. In this way, by simple comparison of the obtained contour up to the actual moment with the desired object contour we do know exactly where there is work left to be done.

We have presented in this paper a method to robustly following an unknown contour and obtaining a precise metric graphic reconstruction of it. Authors are seeking to use the information of both kind of sensors, that is, to fuse the two data sources, in order to obtain an accurate reading of both, the position and the shape, of the unknown object in the robot's workspace.

## ACKNOWLEDGMENT

This work was partially supported by the Generatitat de Catalunya through the multi-arm workstation for the assembly of small electro-mechanical devices project, which is part of the CeRTAP project.

## REFERENCES

- [1] Thomessen T., T. K. Lien, "Robot Control System For Safe And Rapid Programming Of Grinding Applications", 31-st Int. Symp. on Robotics, pp.18-23, 2000.
- [2] H. Kazerooni, M.G. Her, "Robotic deburring of two dimensional parts with unknown geometry", IEEE Int. Symp. on Intelligent Control, 1988.
- [3] S. K. Nayar "Shape recovery methods for visual inspection", Proc. of IEEE Workshop on applications of computer vision, pp 136-145, 1992.
- [4] F. Jatta, R. Adamini, A. Visioli, G. Legnani, "Hybrid force/velocity robot contour tracking: an experimental analysis of friction compensation strategies", Proc. IEEE International Conference on Robotics and Automation, 2002.
- [5] Brown C.M., and D. Terzopoulos, "Real-time computer vision", Cambridge University Press, pp. 232, 1995, ISBN 0 521 47278 4.

- [6] N. Chen, H. Zhang, R. Rink, "Edge tracking using tactile servo", Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.84-89, 1995.
- [7] S. Ahmad, C. N. Lee, "Shape recovery from robot contour-tracking with force feedback", Proceedings of the 1990 IEEE International Conference on Robotics and Automation Barcelona, pp 447-452, 1990.
- [8] J. Baeten, J. De Schutter, "Combined vision/force control at corners in planar robotic contour following", Proc. IEEE/ASME Int. Conference on Advanced Intelligent Mechatronics, 2001.
- [9] C. Kim, J. Park, J. Yi, M. Turk, "Structured light based depth edge detection for object shape recovery", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005.
- [10] M. Bichsel, A.P. Pentland, "A simple algorithm for shape from shading", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 459-465, 1992.
- [11] S. Hegahdaripour, H. Zhang, X. Han, "Investigating of photometric stereo method for 3-D shape recovery from underwater imagery", MTS/IEEE Oceans, pp 1010-1017, 2002.
- [12] Murakami K., and T. Hasegawa, "Tactile Sensing of Edge Direction of an Object with a Soft Fingertip Contact", Proceedings of the 2005 IEEE International Conference on Robotics and Automation Barcelona, Spain, April 2005.
- [13] Liu F., T. Hasegawa, "Reconstruction of Curved Surfaces Using Active Tactile Sensing and surface Normal Information", Proceedings of the 2001 IEEE International Conference on Robotics and Automation Seoul, Korea. May 21-26, 2001.
- [14] G. Beccari, S. Caselli, F. Zanichelli, "Pose-Independent Recognition of Convex Objects from Sparse Tactile Data", Proceedings of IEEE International Conference on Robotics and Automation, pp.3397-3402, 1997.
- [15] Jatta F., A. Adamini, A. Visioli, G. Legnani, "Hybrid Force/Velocity robot contour tracking: an experimental analysis of friction compensation", Proc. Of the IEEE Int. Conf. on Robotics and automation (ICRA), Washington DC, pp. 1723-1728, May 2002.
- [16] K. Nagata, T. Keino, and T. Omata, "Acquisition of an Object Model by Manipulation with a Multifingered Hand", Proc.IEEE/RSJ IROS pp. 1045-1051, 1996.
- [17] Kouji Murakami and Tsutomu Hasegawa, "A New Method of Tactile Sensing Using Fingertip with Soft Skin", Proc. of IEEE/RSJ IROS, pp. 535-540, 2003.
- [18] R.R. Selmik, F.L. Lewis, "Backlash compensation in nonlinear systems using dynamic inversion by neural networks", Asian Journal of Control, 1999.
- [19] O. Faugeras, "Three-Dimensional Computer Vision. A Geometric Viewpoint", The MIT Press, Cambridge, 1993.
- [20] Z. Zhang, "Flexible camera calibration by viewing a plane from unknown orientations", Proc. IEEE International conference on Computer Vision, pp 1150-1153, 1999.
- [21] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses", IEEE Journal of Robotics and Automation, pp 323-344, 1987.