July 1988

Object Exploration Using a Parallel Jaw Gripper

A. Koutsou
University of Pennsylvania

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-88-48.

Object Exploration Using a Parallel Jaw Gripper

Abstract

In this paper we present a system for tactile object exploration. The system is built using a gripper with two parallel fingers, each equipped with a tactile array and a force/torque sensor. We have designed and implemented a set of exploratory procedures for acquiring the following properties: weight, shape, texture, and hardness. The system is successful at extracting these properties from a limited domain of objects. We present a detailed evaluation of the system and the causes of its limitations. The manipulation, motion, and, sensing primitives we have developed in the process of this work could be used for a variety of other tasks, such as model-based recognition, tool manipulation, and assembly.

Comments

University of Pennsylvania Department of Computer and Information Science Technical Report No. MS-CIS-88-48.

OBJECT EXPLORATION USING A PARALLEL JAW GRIPPER

A. Koutsou

MS-CIS-88-48
GRASP LAB 148

Department of Computer and Information Science
School of Engineering and Applied Science
University of Pennsylvania
Philadelphia, PA 19104

July 1988

Acknowledgements: This research was supported in part by DARPA grant NOOO14-85-K-0018, NSF grants MCS-8219196-CER, IRI84-10413-AO2 and U.S. Army grants DAA29-84-K-0061, DAA29-84-9-0027.

Object Exploration Using a Parallel Jaw Gripper

A. Koutsou
Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104

July 7, 1988

Abstract

In this paper we present a system for tactile object exploration. The system is built using a gripper with two parallel fingers, each equipped with a tactile array and a force/torque sensor. We have designed and implemented a set of exploratory procedures for acquiring the following properties: weight, shape, texture, and hardness. The system is successful at extracting these properties from a limited domain of objects. We present a detailed evaluation of the system and the causes of its limitations. The manipulation, motion, and sensing primitives we have developed in the process of this work could be used for a variety of other tasks, such as model-based recognition, tool manipulation, and assembly.

1 Introduction

In this paper we present a system for tactile object exploration. The system is built using a gripper with two parallel fingers, each equipped with a tactile array and a force/torque sensor.

We view object exploration as suggested by psychological studies of [Klatzky and Lederman 86], [Loomis and Lederman 86]. These studies suggest that properties of objects are acquired by stereotypical hand movements called "exploratory procedures". According to their model, the haptic system computes the following properties of objects: a) structural properties: size, shape, and weight, b) surface properties: hardness, elasticity, texture and temperature, and c) functional properties such as part motion.

In our system, we have designed and implemented a set of exploratory procedures for acquiring the following properties: weight, shape, texture, and hardness. For temperature we would require a thermal sensor such as the one described in [Siegel et al 86]. The work presented in this paper follows the work of [Stansfield 86, 87], where a set of exploratory procedures was developed for exploring an object using a single finger. It is important to note that in a robotic system, the implementation of exploratory procedures depend on the geometry of the end effector, its degrees of freedom, and the distribution of sensors. In this respect, some of the issues that we are investigating are the following:

- Advantages/disadvantages of using a gripper vs a probe.
- Advantages/disadvantages of force and torque sensor on the fingers.
- Integration of tactile data with force/torque information.
- Distribution of sensors in a gripper/hand.

Previous work in tactile sensing falls into three major classes: development of tactile sensors, tactile recognition, and tactile exploration. For a review of tactile sensor technology see [Harmon 84].

Although there are many similarities between tactile recognition and exploration, the difference lies in the fact that for recognition a model of the object is assumed to be available. As a consequence, the recognition task aims at extracting and matching some predetermined features. It should also be noted that work in tactile recognition has concentrated on extracting local features related to shape, while other properties, which could only be obtained through tactile information, such as hardness, elasticity, surface texture and temperature, are rarely addressed.

In the area of tactile recognition/exploration, a first distinction we could make is the size of the objects. Objects which are smaller than the tactile pad can be recognized/explored, usually, by obtaining a single tactile image

[Hillis 82]. Larger objects require the temporal integration of more than one tactile image, and thus require the employment of a manipulator. There are four different approaches we can distinguish [Shen et al. 86] in the area of tactile recognition/exploration. The first approach is a probabilistic approach: statistical parameters obtained from a tactile image are compared to reference object statistics [Briot et al 78], [Togai 82]. The second approach is based on a feature extraction methodology: features of an object are extracted from tactile images and compared to stored models [Hillis 82], [Luo et al 84]. The third approach uses positional (kinesthetic) information to compute the coordinates of contact points and deduce the shape of the object [Allen 85]. Finally, the fourth approach, which is only applicable to object recognition, is basically a combinatorial approach: a set of local constraints on identity and location of the objects is developed, and these constraints are pruned in order to eliminate models which are inconsistent with the obtained constraints [Grimson and Lozano-Perez 84], [Ellis 86b], [Browse and Lederman 85], [Grimson 86].

The exploration system presented has been developed under the following two assumptions:

- Some apriori knowledge of object is available. This should include its approximate location and its extent. Currently this information is available from a laser range finder [Tsikos 87].

- The objects to be explored can be grasped by the gripper, i.e. they are smaller than the opening of the gripper at least in one dimension. The objects are, however, larger than the tactile pads in general.

In the next section we give a general overview of the system, both in terms of the hardware configuration and the overall software control structure. In section 3 we present a general description of the implemented exploratory procedures. Section 4 describes the basic primitive actions which are used in the exploratory procedures. In section 5 we describe our methods for analysing force/torque and tactile array data in order to extract the desired properties. In section 6, we evaluate the performance of the system. Finally, in section 7 we summarize our results and describe our current and future work.

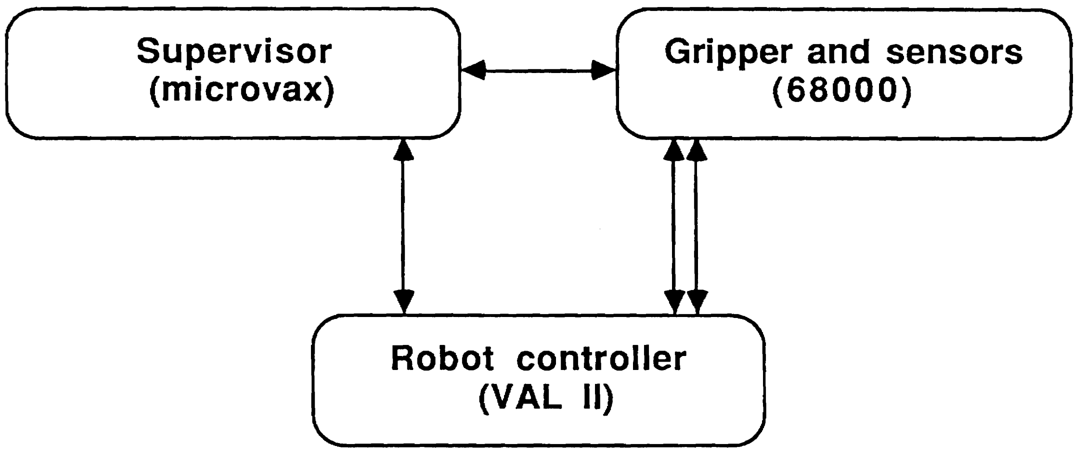


Figure 1: System Configuration

2 System Overview

In this section we present a conceptual overview of the system and its current implementation. We have built the system in a such a way that it is device independent as much as possible. Obviously, the exploratory procedures depend on the geometry of the gripper. But the system is general enough in the sense that it could work with any parallel jaw gripper with tactile sensors.

2.1 System Configuration

Figure 1 shows a block diagram of the system configuration. The system is made up of three components:

1. Robot (PUMA 560) and its controller running VAL-II.
2. LORD gripper with tactile and vector sensing, and its controller (Motorola 68000) running UNIX.
3. Supervisor computer (Microvax-II)

The components communicate via three separate interfaces. The supervisor machine coordinates the whole task.

1. VAX-LEGS interface:
   This is a synchronous interface based on a kermit-like protocol. The supervisor (master) sends manipulation and sensing commands to the gripper and sensors controller (slave). The protocol is responsible for detecting transmission errors, and asking for a packet to be resend, if necessary. On completion of the command, the slave reports to the master the achieved results or, in case of failure, the errors and their cause.

2. VAX-PUMA interface:
   This is based on the supervisor mode of communication of VAL-II. [Izaguirre 86]. The most common way of using the interface is for the supervisor to ask the robot controller to execute a VAL-II program.

3. LEGS-VAL interface:
   This interface is used for pseudo-force servoing of the robot and for termination of a robot motion upon contact with the environment. The robot reads from the parallel port the forces and torques applied to the fingers, and adjusts its motion accordingly.

Below we give some technical data for the gripper and sensors of the LORD Experimental Gripper System (LEGS-I).

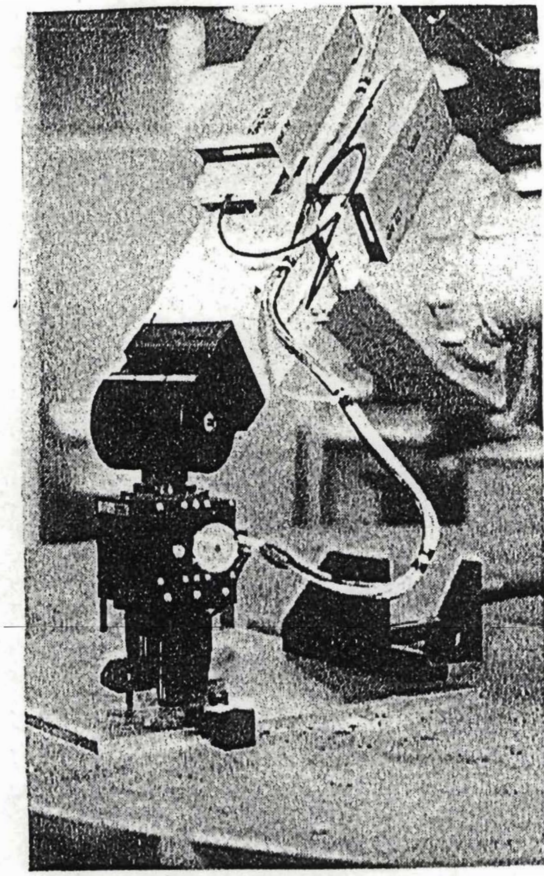
Gripper Data

A schematic of the gripper is shown in Figure 2. The gripper components include: two fingers, each with independent DC servo motor drive; array and vector contact sensing on each finger; a system controller with VME based computer. Some important characteristics of the gripper to note are:

- normal working clamp force: 10lbs (sensor limit)
- maximum clamp force, 20lbs. (sensor limit)
- operating travel: 1 in per finger
- finger separation: 0.125 in min., 2.125 max.

Sensor Data

The sensor we are using is the LTS-200. Some important data about the sensor follows.



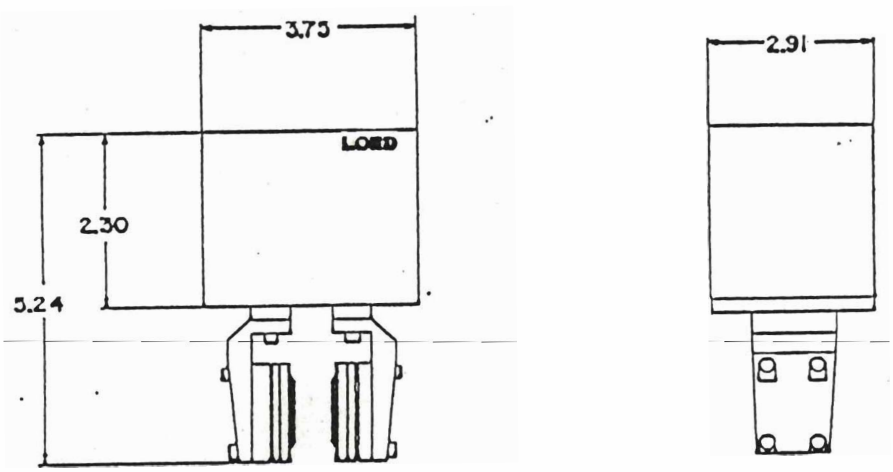


Figure 2: The Lord Experimental Gripper System (LEGS-1).

1. Array sensing

   The tactile pad consists of sensitive sites embedded in an elastomeric touch surface, organized as a 10 x 16 orthogonal array with 0.071 in between sites. The physical size of the pad is 1.8 x 1.1 x 0.83 inches. The deflection at each site is determined using optical sensors. The range of deflection is 0 to 0.030 inches, measured in 16 increments. The site force sensitivity is a few grams.

2. Vector sensing

   The sensor measures force and torque components along and about X, Y, and Z axes of the coordinate system located at the center of the pad. The operating range of the vector sensor is ± 20 lb., ± 35 lb-in, and the resolution is 0.01 lb for force, and 0.01 lb-in for torque.

2.2 Software Overview

Figure 3 shows the conceptual organization of the system software. The software is made up of three layers. At the bottom layer are the robot and gripper controllers. At the next level there are robot / gripper / sensing primitives, which we refer to as actions. At the top level there are the exploratory procedures. Each stage of the exploratory procedure is guided by a model of the object that is built during exploration. At the top level, we need only specify a particular exploratory procedure, and some estimate of the object's size and location, which we get from a 3D laser range finder [Tsikos 87]. This high level specification then translates to a series of actions, which in turn, at the lowest level, is translated into a series of commands to the robot and gripper controllers. Only the software written for the lowest level is device dependent.

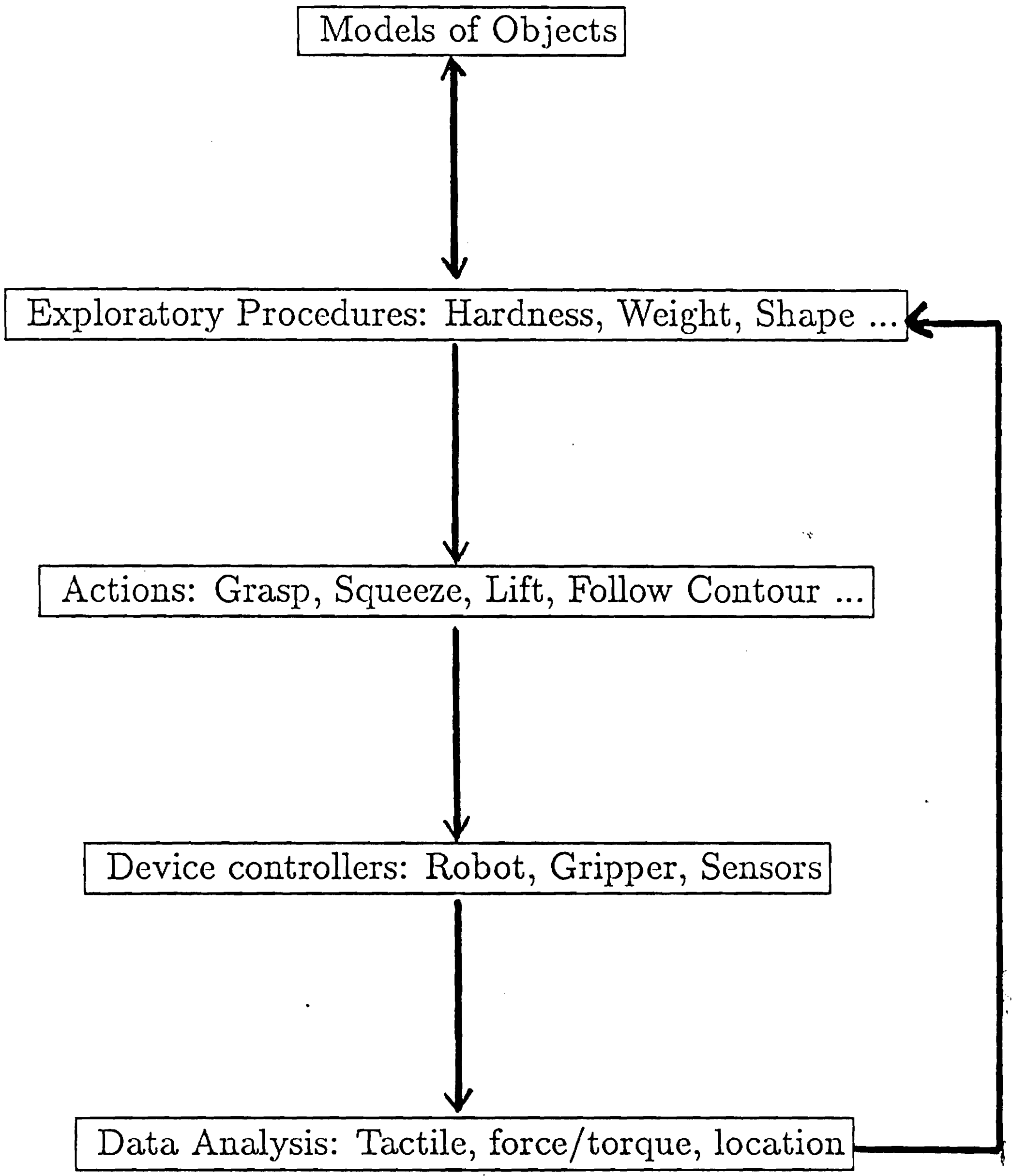


Figure 3: Software control structure.

3 Exploratory Procedures

We have developed and implemented a set of exploratory procedures for obtaining properties of objects. In this section, we give an overall description of each procedure. The underlying primitive actions, methods for analysing the tactile data, and evaluation of the implemented exploratory procedures are discussed in detail in later sections.

3.1 Structural Properties

Weight To determine the weight of an object, we enclose the object, grasp it, lift it, and then measure the forces in the downward direction.

Shape We only extract shape information for reasonably hard objects. To determine the shape of an object, we first enclose the object, grasp it, and obtain tactile images. We then analyse the images to determine what type of surface contact we have. If we cannot detect an edge, we move the gripper up and re-grasp. If we can detect an edge, we extract the orientation of the edge, and follow the edge until all "grippable" surfaces have been explored.

3.2 Surface Properties

Hardness The hardness of an object is a measure of strain/stress. In order to measure hardness, we enclose the object, move the fingers so as to just touch the object, and measure the distance, D1, between the two fingers. Then we grasp the object with force F, and again measure the distance, D2, between the fingers. The hardness is the strain, D1-D2, divided by the stress, F divided by the area of the pad. Clearly, this gives us just an estimate of the hardness of the object, since we do not take into account the elasticity of the pad. We are, however, able to distinguish between such materials as a sponge, a plastic bottle, and a hard block.

Texture We distinguish between smooth and rough surfaces by conducting a statistical analysis of the grey level data in the tactile array.

4 Actions

Exploratory procedures are made up of actions. An action is either a primitive action or a sequence of actions. The actions defined here are specific to the parallel jaw gripper; a set of generalized actions independent of the manipulation tool is undefinable. Following is a description of the actions that have been implemented.

4.1 Motion Primitives

Motion primitives are primitive actions that refer to the motion of the robot. We use three motion primitives: *absolute move*, *relative move*, and *guarded move* (move until force/torque thresholds are exceeded).

An other important primitive which would be useful for manipulation and exploration is compliant motion. Compliant motion is motion which complies with constraints imposed by the geometry of the task. This means that we maintain contact (i.e., forces and torques) while moving. Examples of such motions include sliding and rolling. This would be a useful action for contour following, for example. However, the design principles of the employed tactile pad do not allow for such motion.

4.2 Manipulation primitives

Manipulation primitives are primitive actions that refer to the action of the gripper. There are six major manipulation primitives:

move Move one or two fingers.

> The one degree of freedom linear motion of the gripper can be used for a) moving one finger until it contacts the object or b) moving a grasped object.

center Center the fingers.

> Move the fingers so that the distance between them does not change, but so that the distance from each finger to the edge of the gripper is the same.

squeeze Squeeze object until distance or force exceeds threshold.

This primitive is used in order to detect the elasticity of the object and decide, accordingly, the best way of picking it up. A rough measure of elasticity is obtained by measuring the distance between the fingers on various applied forces. It is also used to adjust the grasping force, in order to obtain better tactile images. It fails when the required force or distance cannot be achieved.

release Release object, while monitoring forces.

By controlling the speed and the opening of the gripper, we can guide the object's release. This can fail if the specified force is very small, due to sensor hysteresis.

grasp Grasp object with specified force.

Given the limited geometry of the gripper (ie, two parallel fingers with one degree of freedom), there is obviously one basic grasping method. However, even within this scheme, there are a number of ways to grasp, and how we want to grasp depends on a number of factors. First, the clamping force used to grasp a glass object ought to be different from the clamping force used to grasp a metal block. Second, in grasping an object, we might have different objectives. We might, for example, want to explore the contours of the object, or we might want to pick it up. Each of these intentions require different applied forces and approach vectors. Finally, in grasping an object, we may or may not have the restriction that the object is not allowed to be moved. So we distinguish between centered and off-centered grasping: in off-centered grasping, we move one finger until it touches the object, and then the other finger until some specified clamping force is achieved; in centered grasping, both fingers move simultaneously.

Given these considerations, the grasping primitive allows the user to specify the required forces at the fingers (with the required degree of precision), as well as centered or off-centered grasping.

Re-orient Re-orient a grasped object such that the center of gravity of the object lies on the downward axis.

Suppose we have a tight grip on an object such that its center of gravity is not on the downward axis. (Imagine holding a rectangular block at one end of the longer side so that the longer side extends in the horizontal direction.) We can detect that this is the case by the measured torques on the fingers. This primitive reorients the object in the hand by reducing momentarily the clamping force and then increasing it again to the original force.

4.3 Sensing Primitives

Sensing primitives are primitive actions that refer to operations of the sensor and are specific to LTS-200 sensors. These include:

- Scan tactile array pads.
- Scan force/torque vectors.
- Set threshold for forces/torques.
- Continuous scan until threshold exceeded.
- Differential scan of array pads.

4.4 Combined Actions

The following actions combine two or more primitive actions.

enclose Enclose an object between the two fingers of the gripper.

We use information about the object size and location to enclose an object. This information is currently acquired from a 3-D laser range finder. The errors in the measured size and location would be of no consequence if the object is small relative to the size of the maximum opening of the gripper: by moving the gripper above the object and then lowering the gripper, we would successfully enclose the object. However, if the object is relatively large, an error in position, orientation, or size would result in a collision between the gripper and the object. Therefore, we use a combination of guarded move, move, and re-orient in order to grasp an object.

We can enclose objects of maximum width 38 mm. For the widest possible object, we can can accomplish the enclose action given that the error in orientation is less than 6 degrees and the error in position is less than 5 mm.

pick up Enclose, grasp and move the robot up.

follow contour Move the gripper along the contour of an object.

We determine the direction of motion according to the edges detected by the sensor. To follow the contour, we actually grasp, open the gripper, move the gripper, and re-grasp. The procedure ends when one of the following happens:

1. We meet an obstacle.
2. The object no longer fits between the two fingers of the gripper.
3. We return to the starting position (in the case of a closed curve).

This action involves "active" sensing, in the sense that at each step we determine what to do next based on the tactile information we have just acquired. We have two sources of information, i.e., the two tactile arrays. At any point, only one finger is considered to be the active one. One advantage of using two sensors is that if we fail to obtain a recognizable feature using the data from the active sensor, we can make the non-active sensor the active one, and continue the exploration. Another advantage of using two sensors is the fact that we can easily detect asymmetry, simply by comparing the features extracted from the two tactile arrays.

The problems/difficulties in this procedure arise when dealing with non-planar surfaces. With the gripper we are currently using, what we are specifically interested in finding out is if the curvature of the surface is at least as large as that of a cylinder, since the procedure for following the contours for such surfaces will be different from the procedure for following the contours of a surface that is nearer to planar. For a planar, or almost planar surface, the gripper is moved in a plane parallel to the surface tangent. In the case of an almost planar surface, this motion will result in a partial contact of the tactile arrays with the surface and eventually to non-contact at all, since the two fingers are always parallel. At that point we have to rely on force/torque information exclusively for feature extraction. The procedure for following a cylindrical-like surface is to repeatedly rotate the gripper 45 degrees along the edge until we reach the starting position.

One thing we conclude from our experiments in contour following is that the gripper we are using imposes a lot of constraints on the ways we can position the tactile arrays with respect to the object. A single probe or multi-fingered hand with retractable fingers is better suited for this task.

5 Data Analysis

In this section we describe how we analyse the data acquired during the exploratory procedures in order to extract object properties. At the low level, the available data is the following:

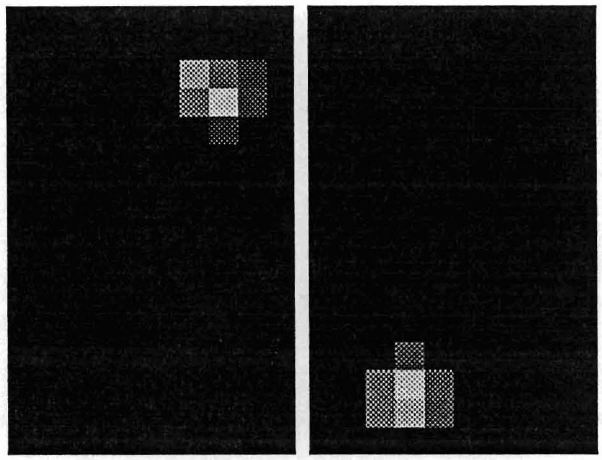
- position and orientation of the robot;
- position of the fingers with respect to the gripper;
- force/torque information on the fingers;
- tactile array images.

5.1 Force/Torque Analysis

There are several approaches to using force/torque information. In one approach, force/torque information is used essentially for positioning the gripper. For example, we could use the measured torques to position the tactile pad flush against a planar surface in order to obtain surface information such as bounding edges and texture. Or we could use force/torque information to perform motions such as guarded and compliant motions.

In another approach forces and torques are used to extract contact information [Salisbury 85]. What has not yet been sufficiently researched is the use of both the tactile array and force/torque data for finding contact information and therefore feature extraction. This would be useful in two instances. First, we could use such a scheme to check that the features we extracted through tactile information are consistent with force/torque data. Second, there might be cases where we cannot place the tactile pad on the object we wish to explore, either because of the constraints imposed by the environment of those imposed by the gripper itself. In these cases, we can either use the force/torque data extracted from placing the sides of the finger on the object or we can use force/torque data acquired from a probe held between the two fingers.

In this study we use force/torque data in a number of ways. First, we use it in positioning operations such as guarded motion and grasping. Second, we use forces and torques measured on the sides of the fingers in the enclose action, as discussed earlier. In addition, we use force/torque data to verify



a b

Figure 4: Point contact - In both (a) and (b) the tactile array is in contact with a sphere of 2.5 mm radius. The location of the point contact is determined both by analysis of the tactile array, and by analysis of the force/torque information. The distance between the location of the points found by the two methods is in (a) 1.23mm, and in (b) 1.13 mm. Note that the distance between sites is 1.75mm.

the data in the tactile arrays in the case where we have a point or line contact. Figure 4 shows two tactile arrays obtained by contact with a spherical surface of 2.5 mm radius. The coordinates of the contact point are calculated from the force/torque measurements using the method described in [Salisbury 85]. The location of the contact point is also computed from an analysis of the tactile image. We have compared the results obtained by these two methods, and we have found that the average distance between the two estimated points is 2.10mm. This error is greater than the expected error due to sensor noise (see [Salisbury 85]), which would be less than 1 mm for normal forces greater than 0.8 lbs. We have attributed this discrepancy to the elasticity of the sensor array pad. And finally, we have begun a series of experiments to study the force/torque data obtained when handling tools such as spatulas and probes.

5.2 Tactile Image Analysis

In this section we discuss the static image analysis we perform on the tactile array for each finger in order to extract the following information:

- Type of contact (point, partial contact, full contact, ridge).
- Surface type (planar vs. curved).
- Surface texture (smooth vs. coarse).
- Feature extraction (edges, corners, holes).

Tactile image analysis differs from image analysis for computer vision in several ways. First, a tactile array, because it measures deflection, directly gives three-dimensional information. Also, in a tactile array, the data is sparse, which makes image analysis computationally simpler, but the information is less detailed. This means that the vast literature on computer vision is not directly applicable to the problem of analysing tactile arrays.

There are two approaches to improving tactile image analysis. We could ensure that we get the best possible tactile image, by carefully monitoring the orientation of the tactile pad to the surface being explored. This would involve re-adjusting the orientation whenever we found a "bad" image, a time-consuming procedure. We could also invest effort in the feature detection algorithm, so that it could analyse data with more imperfections. This is algorithmically complex. The robustness of the system described relies on a combination of these two approaches.

The rest of this section details the tactile image analysis methods that have been implemented.

5.2.1 Feature extraction

We categorize the type of contact encountered and extract the features of the object according to (a) the proportion of the tactile pad which registers force information (b) the elongation of the area in contact, i.e. the ratio between major and minor axes. (c) the centroid of the area in contact, and (d) the orientation of the area in contact.

We also compute the first- and second-order moments of the regions. But we have found that because of the sparse tactile data and the small grey-scale range, the features we extract through moment information are most often erroneous.

We have extended the work of [Stansfield 87] in this area in order to extract hole information, and the bounding polylines of the object. The bounding polylines of the object are found by applying the fit-and-split method on the boundary points of the contact area.

The different types of contact and the corresponding extracted features are described below:

- Full contact: This means that almost all the tactile sites are registering force information. In this case, we check whether or not there is a hole. Figure 5 shows full contact with a planar surface having a hole of radius 3.5mm. From the analysis of the tactile data, we obtain the centroid and the area of the hole. Theoretically, the smallest hole which can be detected has radius equal to the distance between adjacent sites (1.7mm). The interdependence of adjacent sites, however, as well as possible imperfections of the surface of the object impose additional limitations. Through a number of experiments, we have concluded that the minimum hole, in terms of its radius, which can be reliably detected is a hole with r = 3.4mm.

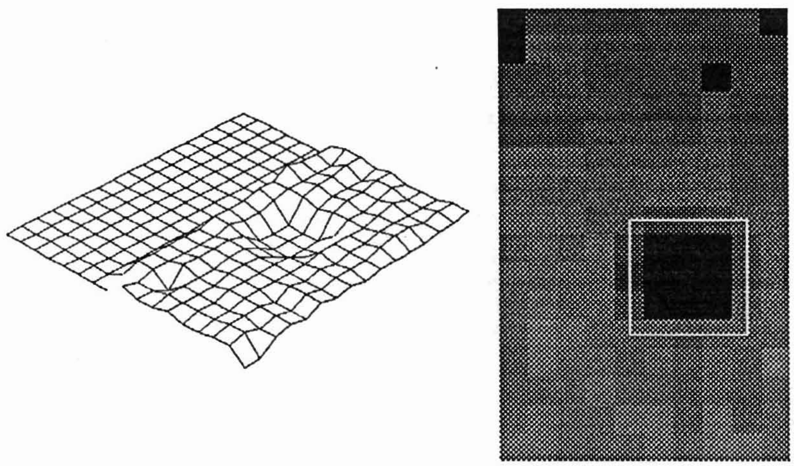
- Partial contact: A substantial part of the sensor pad is in contact with the object. In this case we further categorize the contact as:

  - Boundary edge contact: An object is partially in contact with the pad, and the boundary between contact/non-contact is a single edge. The location, orientation and extent of the edge is recorded. See Figure 6.

  - Corner contact: An object is partially in contact with the pad, and the boundary between contact/non-contact is a corner. See Figure 7.

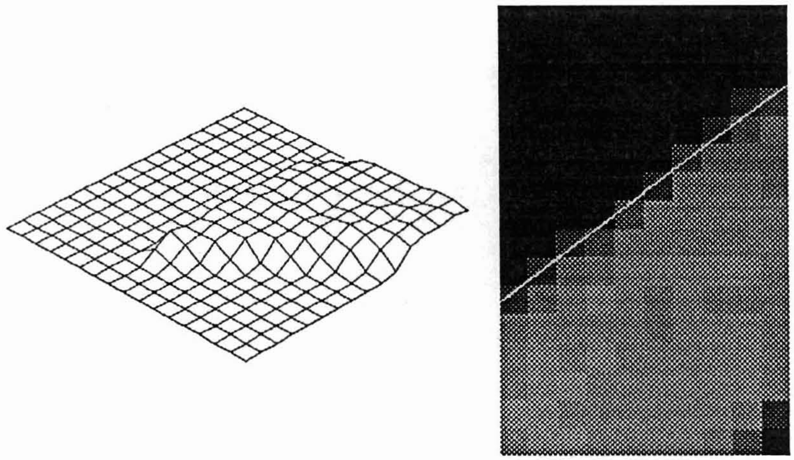
  - General partial contact: This case takes care of the remaining possibilities. We extract the bounding edges and holes. As an example, Figure 8 shows the tactile image of washer (4mm radius) and the detected bounding edges.

- Point: Only a small proportion of the pad is in contact with the object, and the elongation of the area in contact is small. Point contact can



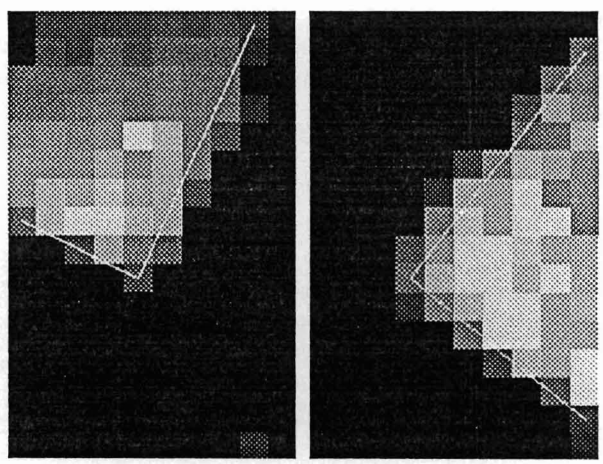
a b

Figure 5: Full contact with a planar wooden surface containing a hole of radius 3.5 mm. In (a) the deflections at each site of the array are shown on the right part of the 16 X 16 grid. In (b) the intensity of each grid element is proportional to the deflection at each site. The computed outline of the hole is shown.



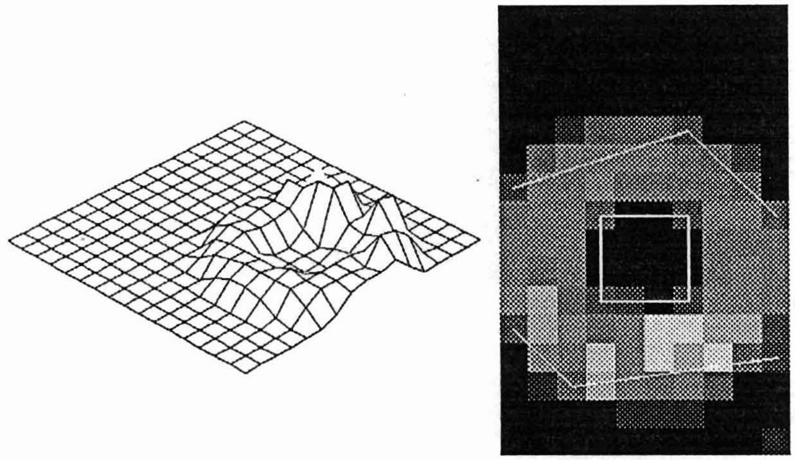
a b

Figure 6: Bounding edge of a block – (b) shows the computed edge of the block.



a b

Figure 7: Corner Contact - In (a) and (b) a corner is identified at the boundary of the object.



a b

Figure 8: Tactile images of washer of radius 4mm – (b) shows the found bounding edges of the washer.

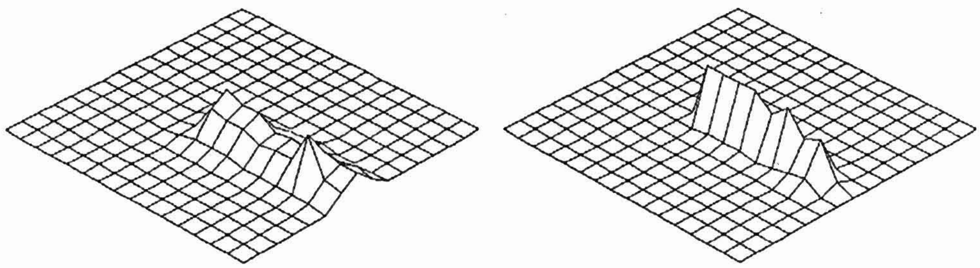
result from a vertex contact or spherical contact. We measure location of point, and spread of forces. See Figure 4.

- Ridge: This type of contact is characterized by a high elongation, and high intensity along the major axis of the contact area. The situation can result from an edge or a cylindrical contact. Figure 9 illustrates the difference between these two different cases.

  We are able to distinguish between cylinder and edge contact, provided that the radius of the cylinder is greater than 1mm. In Section 5.2.2 we discuss methods discriminating between different types of surfaces.

  In the case of ridge contact, we measure the length, thickness, and direction (major axis) of the ridge (see Figure 10).

- Multiple contacts: Disconnected regions of the tactile array show positive deflection. Currently, we don't analyze the tactile data any further.



a b

Figure 9: Ridge contact - In (a) the tactile array is contact with a cylinder of radius 20mm. In (b) the array is in contact with an edge.

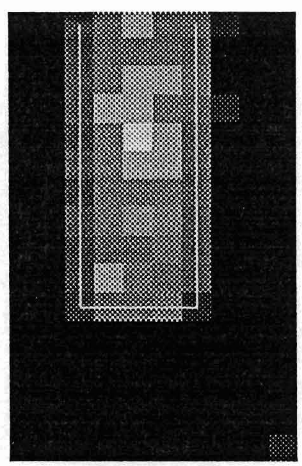


Figure 10: Ridge contact – Tactile image of a cylinder (radius = 16 mm). The bounding edges of the ridge are shown.

5.2.2 Surface Type

We are able to classify surfaces as either planar, cylindrical, or spherical. Our method will only give accurate results if we are dealing with smooth surfaces and we apply sufficient force (6 lbs) to obtain large enough deflections in the tactile pads.

The analysis is based on the grey scale values of the tactile array. We assume that the contacted surface is a plane, and then fit a plane to the surface using linear regression. If the variance is larger than some threshold value we have determined through experimentation, we conclude that the surface is not planar. If the surface is determined to be non-planar, and the type of contact is a ridge, then we try to fit a cylinder to data. If we cannot fit a cylinder, we conclude that the tactile image is that of an edge. Likewise, if the surface is determined to be non-planar and we have a point contact, we try to fit a spherical section to the data. If we cannot fit a spherical section, we conclude that the tactile image is that of a vertex.

We might want to obtain more accurate surface descriptions, if our goal were to build a precise geometric model of the object, or to recognize the object in a domain of similar objects. To do this, we would want to construct a model of the surface using surface patches, or B-splines, or Bezier surfaces [Overton 84], [Shen et al. 86].

5.2.3 Surface Texture

We are able to distinguish smooth vs. rough surfaces. To determine this, we conduct a statistical analysis of the grey level data. For experimental results see [Stansfield 86]. For determining more detailed texture properties, a higher spatial resolution is necessary [Ellis 86a], or a dynamic type of sensor, which recognizes changes in forces as it's moving on a surface.

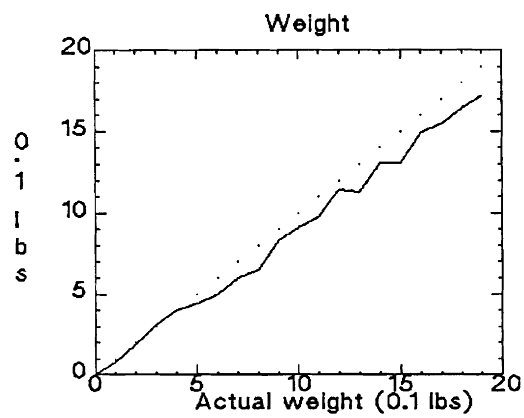


Figure 11: Measured vs. actual weight of object held

6 System Evaluation

6.1 Weight

We have measured the weight of various objects by grasping the object, lifting it, orienting the robot so that the y-axes of the sensors point downwards, and summing the forces on the y axes. Figure 11 shows measured weight vs. actual weight of an object held by the gripper with clamping force 2.5 lbs.

Range of weights: 0 - 8 lbs (sensor limit).

Accuracy: ± 0.1 lbs

Independent of grasping force.

6.2 Hardness

We measure the hardness of an object by measuring how much the object deforms when normal forces are applied. Figure 12 shows the difference in measured object width with various clamping forces. In the case of a metal block, we would expect a constant width, irrespectively of applied forces. But, as can be seen from the figure, this is not the case. What we are really

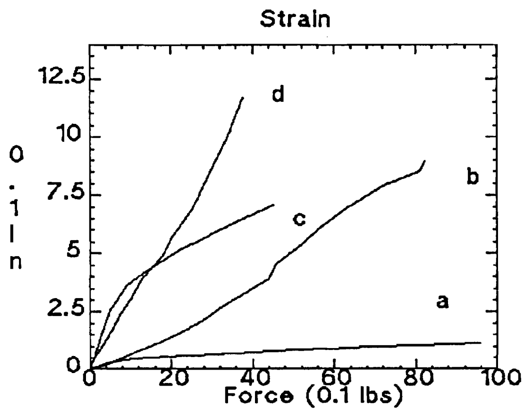


Figure 12: Hardness of objects. The graph shows the change in object width for different clamping forces. (a) a metal block; (b) a plastic bottle (c) a cardboard box; bottle; (d) a sponge.

measuring is the elasticity of the tactile pad. The strain/stress factors for the four objects are: for the metal box, 0.01 – 0.03; for the plastic bottle, 0.1 – 0.18; for the cardboard box, 0.13 – 0.44; and for the sponge, 0.26 – 0.31. We can classify, therefore, an object as hard if the strain/stress factor is less than 0.05, and as very elastic if it is more than 0.2.

6.3 Contour Following

We have explored the contour of objects of small weight (less than 0.5 lb). Figures 13 – 15 show the edges found by analysing the tactile arrays. Figure 13 shows the edges of a block, figure 14 the edges of an arch, and figure 15 the edges of a cylinder.

Figures 13 and 14 show the objects placed at 3 different positions. The initial grasping location was provided by a laser scanner. The number of grasps performed were: for the block, 15-17; for the arch, 17-20; and for the cylinder, 13-16.

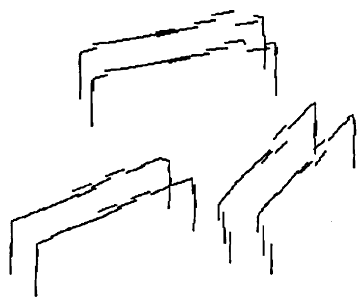


Figure 13: Edges found for a block of dimensions 140 x 30 x 60 mm, 0.2lbs

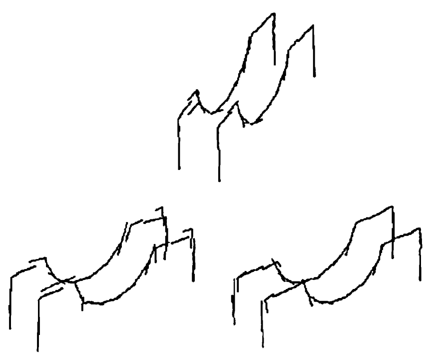


Figure 14: Edges found for an arch of dimesions 140 x 30 x 60 mm, 0.15 lbs

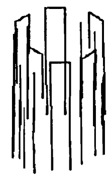


Figure 15: Edges found for a cylinder of 20mm radius, 50mm height

As can be seen from the figures, the obtained segments of the edges fall close to the actual edges. But how close? In the next sections we evaluate the results of the contour following procedure. In particular, we will address the following questions:

- How do results vary for different runs on the same object?
- How much does the object move during tactile exploration? Is this a limiting factor in extracting shape information?
- What is the range of features that can be detected?
- Can we construct a model of the object?
- How close are the found edges to the actual edges? How close are the found edges to a fitted model?

6.3.1 Statistical Analysis of the Contour Following Procedure

The questions which we will address in this section concern the reliability, repeatability, and accuracy of the shape information provided by the tactile

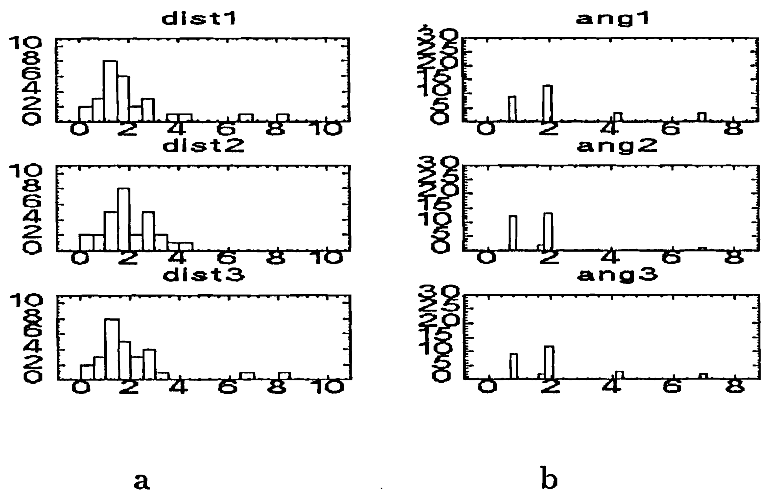


Figure 16: Deviations in (a) distance (mm) and (b) angle (degrees) between *actual* edges and edge segments obtained thought repeated tactile exploration of the same block

exploration system.

In order to perform this analysis, we explored the same object at the same initial location three times. The explored object was the simplest possible, i.e., a block. The edges of the block were straight with no observable defects.

We first compared the obtained segments of the edges with the actual edges. The segments of the edges are represented by their two end points. We computed the distance and the angle between every segment and the actual edge on which it should lie. The histograms in Figure 16 show the computed distance and angles for each of the three measurements.

As can be seen from the figure, the errors in angle are quantized. This is a result of the resolution of the tactile array. Taking into account the dimensions of the array and the number of sites, the minimum angle between two edges that can be detected is 3.49 degrees.

We compared the means and the variances of the three cases at the 0.05 significance level, using the Bartlett test, and we found that they are the same. We conclude, therefore, that the three different experiments yielded similar error distributions, and the shape exploration procedure is, thus, consistent and repeatable.

For the case of distances, a 95% confidence interval for the mean of the distribution is 1.70 – 2.37, and the estimated variance is 2.45. For the case of angles, a 95% confidence interval for the mean is 1.65 – 2.36, the estimated variance is 2.745. We conclude that the average error in distance is approximately 2mm, and the average error in angle is approximately 2 degrees.

The main source of error in the location of the found edge segments is attributed to the motion of the object during exploration, since the positional accuracy of the robot and the gripper, as well as the resolution of the tactile array would yield errors below the measured averages. This source of error could be eliminated either by holding the object by another gripper or vice, or by choosing to explore heavy objects. We chose, however not to restrict the motion of the object, because we were interested in assessing the effects of the interaction between the robot/gripper and the object, which is one of the major characterists of active tactile exploration.

6.3.2 Contour Models

Having compared the acquired edge segments with the actual edges, the issue that we address here is how to obtain some model of the actual contours. As a first step, we concentrated on straight edges. We rejected the least squares fitting method on the grounds that a piece of erroneous information, such as an edge segment obtained when the object is tilted would have resulted in a modelled edge very different from the actual edge. What we basically want to do, is filter out segment edges that differ significantly from the rest, unless there is evidence that they constitute a feature of the object. The method we followed is summarized by the following steps:

1. Divide all the found edge segments into sets such that within a group the angles and the distances between all segment edges are below some threshold values. The threshold values are chosen according to our knowledge of error distributions.

2. For each set whose cardinality is greater than one, find a nominal segment, i.e., a segment for which the sum of distances and angles between it and other segments in the same set is minimal. From this segment, we obtain the cosines of the fitted line. The endpoints of the line are found by projecting the endpoints of the segments on the line, and choosing local minima and maxima.

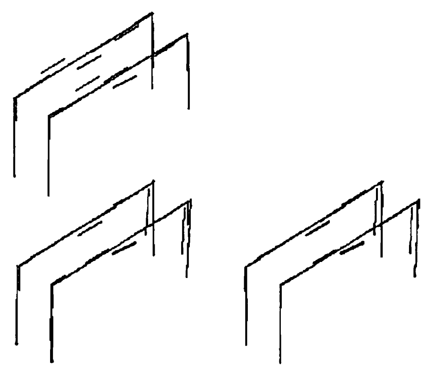
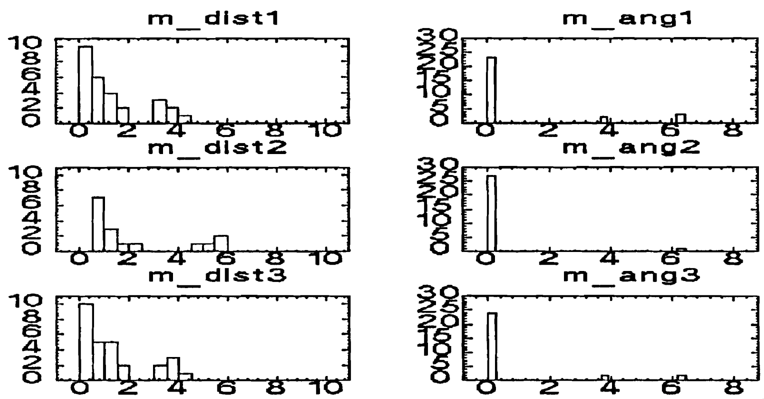


Figure 17: Edges found for the same block in three instances, and fitted lines

The three straight line models obtained by this method for the three cases of exploration of the block are shown in Figure 17. We compared the edge segments to the fitted lines by means of computing the distances and the angles. The histograms in Figure 18, show the computed deviations. As in the previous section, we compared the distribution of errors in the three cases, and we found that the means and the variances are the same, at the 0.05 significance level. The modelling process gives, therefore, consistent results. In the case of distances, a 95% confidence interval for the mean is 0.97 – 1.61, and the estimated variance is 2.35, while in the case of angles, the interval for the mean is 0.27 – 1.03, and the variance is 3.17.

Finally, we compared the obtained models of the edges with the actual edges in terms of distances between the edges, angles between the edges, and difference in length. The computed results are shown in Figure 19. The 95% confidence intervals are (a) for the mean error in distance: 1.19 – 2.08 mm, (b) for the mean error in angle: 0.97 – 2.03 degrees, and (c) for the mean error in length: 0.97 - 2.03 mm. The estimated variances are 0.92, 1.32, and 1.48 respectively.



a b

Figure 18: Deviations in (a) distance (mm) and (b) angle (degrees) between *fitted* edges and edge segments obtained thought repeated tactile exploration of the same block

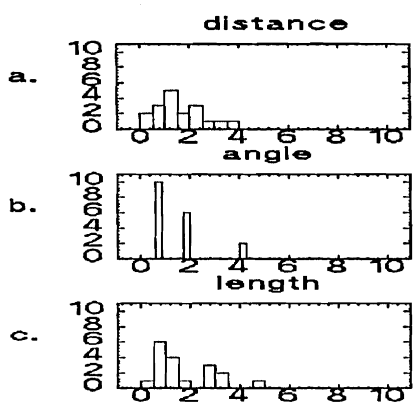


Figure 19: Deviations in (a) distance (mm), (b) angle (degrees), and (c) length (mm) between *actual* edges and edges *fitted* to the edge segments obtained thought repeated tactile exploration of the same block

In summary, we can conclude that the contour following procedure gives reliable and consistent results. Even though the objects were free to move, we were able to construct an edge model of the object within an accuracy of approximately 2mm and 2 degrees. Small imperfections on the contour of an object, such as a protrusion of less than 2mm, will not, however be detected as such.

7 Conclusions

We have described a system for tactile exploration of objects using a parallel jaw gripper. We have implemented a set of exploratory procedures to be used with a two fingered gripper in order to extract object properties such as weight, shape, hardness, and texture. The most complex part of this research has been the extraction of shape information.

The system has been shown to be successful at extracting the above properties from a limited domain of objects. We have quantitatively defined the range of features that can be reliably detected. And we have presented a detailed analysis of the causes of the system's limitations in terms of error and range.

The implementation of the exploratory procedures in any system is dependent on the employed hand. The basic advantage of the two-fingered gripper we have used as compared to a single finger is the fact that we can manipulate an object with the gripper. We can therefore extract information about an object's weight, center of gravity, and part motion, which we have not yet investigated. In addition, we can use one of the fingers to stabilize the object, and we can therefore explore objects which are relatively light without moving them. We have also demonstrated how it can be useful to have two tactile arrays. We can use the data in one array to compensate for poor data in the other. Using two tactile sensors also means that we can explore the object using fewer motions, thereby obtaining the global shape of an object more quickly.

Yet the gripper has posed its problems. Its constrained geometry in terms of the degrees of freedom of motion and the opening of the fingers means that only a limited domain of objects can be explored. In addition, exploration methods are limited in at least two ways. First, it is not always possible to place the tactile pads at the desired location or orientation. And second, due to the design of the sensor and its tendency to "blister", it is impossible to perform compliant motion, which would be the optimal way to follow a contour. These constraints make contour following, and hence extraction of local shape, a difficult task. An articulated hand is obviously more suited to the task of object exploration. Given the current tactile sensor technology, the articulated hand should have force/torque sensors on some fingers which would be used when manipulating an object, and tactile array sensors on other fingers or the palm, which would be used for feature extraction [Ulrich

88]. Obviously, however, this involves more complex control and sensory integration issues.

The manipulation, motion, and sensing primitives we have developed in the process of this work could be used for a variety of other tasks, such as model-based recognition, tool manipulation, and assembly. For assembly tasks, we could obtain precise information about the location/orientation of the part. This information can be extracted either as the part is moved to some desired location, or during some initial verification stage. We are currently investigating the use of tactile sensing for tool manipulation and dis-assembly.

References

[Allen 85] P. Allen. *Object Recognition Using Vision and Touch.* PhD Thesis, University of Pennsylvania, 1985.

[Bajcsy 84] R. Bajcsy. What Can We Learn From One Finger Experiments? In M. Brady and R. Paul, editors, *The First International Symposium on Robotics Research,* The MIT press, Cambridge, Massachussets, 1984.

[Briot et al. 78] M. Briot and M. Renaud and Z. Stojiljkovic. An approach to spatial pattern recognition of solid objects. *IEEE Trans. on Systems, Man and Cybernetics,* SMC-8:(9):690-694, 1978.

[Browse and Lederman 85] R. Browse and S. Lederman. Feature-Based Robotic Tactile Perception. In *Proceedings of Computer Aided Technology,* 1985.

[Ellis 84] R.E. Ellis. Extraction of Tactile Features by Passive and Active Sensing. In *Proceedings of SPIE Vol. 521 Intelligent Robots and Computer Vision,* Cambridge, Massachusetts, November 1984.

[Ellis 86a] R.E. Ellis. A Multiple-Scale Measure of Static Tactile Texture. In *Proc. IEEE Conf. Robotics and Automation,* (2):1280-1285, 1986.

[Ellis 86b] R.E. Ellis. *A Tactile Recognition Strategy for Planar Objects.* COINS Technical Report 86-32, Department of Computer and Information Science, University of Massachusetts, Amherst, 1986.

[Grimson and Lozano-Perez 84] W.E.L. Grimson and T. Lozano-Perez. Model Based Recognition and Localization from Sparse Range or Tactile Data. *International Journal of Robotics Research.* 3(3), 1984.

[Grimson 86] W.E.L. Grimson. Disambiguating Sensory Interpretations Using Minimal Sets of Sensory Data. In *Proc. IEEE Conf. Robotics and Automation,* (1):286-292, 1986.

[Harmon 84] L. Harmon. Tactile Sensing for Robots. In Brady et al., editors, *Robotics and Artificial Intelligence,* NATO ASI Series, Springer-Verlag Berlin Heidelberg, 1984.

[Hillis 82] W.D. Hillis. A High-Resolution Touch Sensor. *International Journal of Robotics Research,* 1(2):33-44, 1982.

[Klatzky and Lederman 85] R.L. Klatzky and S.J. Lederman. *Hand Movements: A Window into Haptic Object Recognition,* Cognitive Science Technical Report 8606, University of California at Santa Barbara, 1985.

[Loomis and Lederman 86] J.M. Loomis and S.J. Lederman. Tactual Perception. In K. Boff, L. Kaufman and J. Thomas, editors, *Handbook of Perception and Human Performance,* Wiley, New York, 1986.

[Luo et al. 84] R.C. Luo, W.H. Tsai, and J.C. Lin. Object Recognition with Combined Tactile and Visual Information. In A. Pugh, editor, *Proceedings of the 4th Iternational Conference on Robot Vision and Sensory Control,* IFS (Publications) Ltd, UK North Holland, 1984.

[Overton 84] K. Overton. *The Acquisition, Processing, and Use of Tactile Sensor Data in Robot Control.* COINS Technical Report 84-08, University of Massachusetts, Amherst, Massachusetts, 1984.

[Salisbury 85] J.K. Salisbury. Interpretation of Contact Geometries from Force Measurements. In M.T. Mason and J.K. Salisbury, *Robot Hands and the Mechanics of Manipulation,* The MIT Press, 1985.

[Shen et al. 86] J.T. Shen, K.A. Lopard, and C.W. Thomas. An Active Touch Processing System. In *Proc. of Sensors '86* Detroit, Michigan, November 11-13, 1986.

[Siegel et al. 86] D. Siegel, I. Garabieta. and J.M. Hollerbach. An Integrated Tactile and Thermal Sensor. In *Proc. IEEE Conf. Robotics and Automation,* (2):1286-1291, 1986.

[Stansfield 86] S. Stansfield. Primitives, Features and Exploratory Procedures: Building a Robot Tactile Perception System. In *Proc. IEEE Conf. Robotics and Automation,* San Francisco, 1986.

[Stansfield 87] S. Stansfield. *Visually guided haptic object exploration.* PhD Thesis, University of Pennsylvania, 1987.

[Togai 82] M. Togai. *Pattern recognition schemes for a touch-sensor array based on moment-invariant and circle search methods.* Technical Report, Robotics-EE-82-03, Duke University Robotics Research Cooperative Research Program, 1982.

[Tsikos 87] G. Tsikos. Segmentation of 3-D Scenes using Multi-Modal Interaction Between Machine Vision and Programmable, Mechanical Scene Manipulation. Ph.D. thesis, University of Pennsylvania, 1987.

[Ulrich 88] N. Ulrich, R. Paul, and R. Bajcsy. A Medium-Complexity Compliant End Effector. In *Proc. IEEE Conf. Robotics and Automation,* Philadelphia, 1988.