Current Research in Robotics and Automation

Four overviews provide a sampling of current research in robotics. Included are reports from Columbia University, Sandia National Laboratories, Cornell University, and the University of Southern California.

An Intelligent Grasping System

Peter Allen, Paul Michelman, and Kenneth Roberts
Department of Computer Science, Columbia University, New York, NY 10027

Work in dextrous manipulation at Columbia University's Computer Science Department focuses on building a comprehensive grasping environment capable of performing tasks such as locating moving objects and picking them up, manipulating man-made objects such as tools, and recognizing unknown objects through touch. In addition, we are designing an integrated programming environment that will allow grasping and grasping primitives within an overall robotic control and programming system that includes dextrous hands, vision sensors, and multiple-degree-of-freedom manipulators.

Our current system contains the following components:

- a set of low-level system primitives that serve as the basis for the control of the hand,
- a true hand/arm system with many degrees of freedom formed by mounting the hand on a robotic manipulator, a Puma 560,
- integrated tactile sensors on the fingertips that provide force-sensitive responses and Cartesian position information, and
- sensing primitives that use joint position, tendon force, and tactile array sensing in a number of grasping and manipulation tasks.

System overview. The system we have built consists of a Utah-MIT hand[1] attached to a Puma 560 manipulator (Figure 1). The hand has four fingers, each with four degrees of freedom. It resembles the human hand in size and shape, but it lacks a number of features humans find very useful. In particular, it has no palmar degree of freedom (closing of the palm), and the thumb is placed directly opposite the other three fingers, with all fingers identical in size. The hand has joint-position sensors that yield joint-angle data and tendon-force sensors that measure forces on each of the two tendons (extensor and flexor) controlling a joint. The Puma adds six degrees of freedom to the system (three translation parameters to move the hand in space and three rotational parameters to orient the hand), yielding a 22-degree-of-freedom system. The system is controlled by an intelligent, high-level controller that links arm, hand, and finger movements with the feedback sensing of joint positions, tendon forces, and tactile responses on the fingers.

Figure 2 shows the system's hardware structure. The high-level control resides in a Sun-3 processor. The Sun serves as the central controller and has access to a full Unix-based system for program development and debugging as well as a set of window-based utilities to allow graphical output and display of the system's various states. The hand is controlled by an analog controller commanded through D/A boards from a dedicated 68020 system. The Sun is capable of downloading and executing code on the 68020 and can communicate with it through a shared memory interface.[2] The low-level system primitives to control the hand include the ability to read and write joint positions and tendon forces, and forward and inverse kinematics to allow both joint-space and Cartesian-space operation of the hand. The joint-position and tendon-force sensors can also be monitored at high sampling rates to monitor the hand's operation in real time, for both movement and contact grasping.

The tactile sensing system is controlled by another dedicated 68020 that monitors the forces on each of the sensor pads. The tactile sensors we are using consist of 16×16 grids of piezoresistive polymeric material that are conformable to the finger's shape.[3] They are manufactured by sandwiching the polymeric material between two pliable sheets of Kapton that contain electrical etching. The application of forces on the pads provides an increased electrical flow channel between the two sheets as the material within is compressed. Results with this sensor have been good, particularly with respect to signal isolation. The sensors are monitored by a separate 68020 that is responsible for low-level tactile processing, including A/D conversion, thresholding, and signal normalization. A number of tactile filters, including edge and line detection, smoothing, and moment analysis, have also been implemented.

The control of the Puma arm is through a serial link that allows the Sun to act as the Puma system host (host control option). This allows the system to use the full range of robotic arm commands. These include movement primitives, asynchronous interrupt capability, and the ability to establish arbitrary coordinate frames such as the hand coordinate system. It also provides a global coordinate system in which the tasks can take place.

Applications. The system has performed a number of grasping tasks, including pick and place operations, extraction of circuit boards from card cages, pouring of liquids from pitchers, and removing light bulbs from sockets. These tasks have been programmed using Dial, a graphical animation language developed by Steven Feiner.[4] This diagrammatic language allows representation of parallel processes in a compact graphical "time line" and has been used for animation of graphical displays. We transported it to the robotics domain and have exploited its ability to express parallel operation of robotic devices. Dial also provides a convenient way to implement

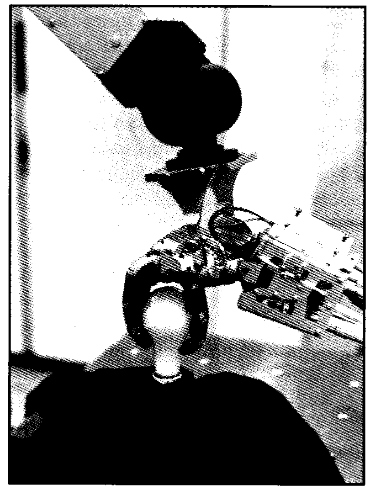

Figure 1. Utah/MIT hand mounted on Puma 560.

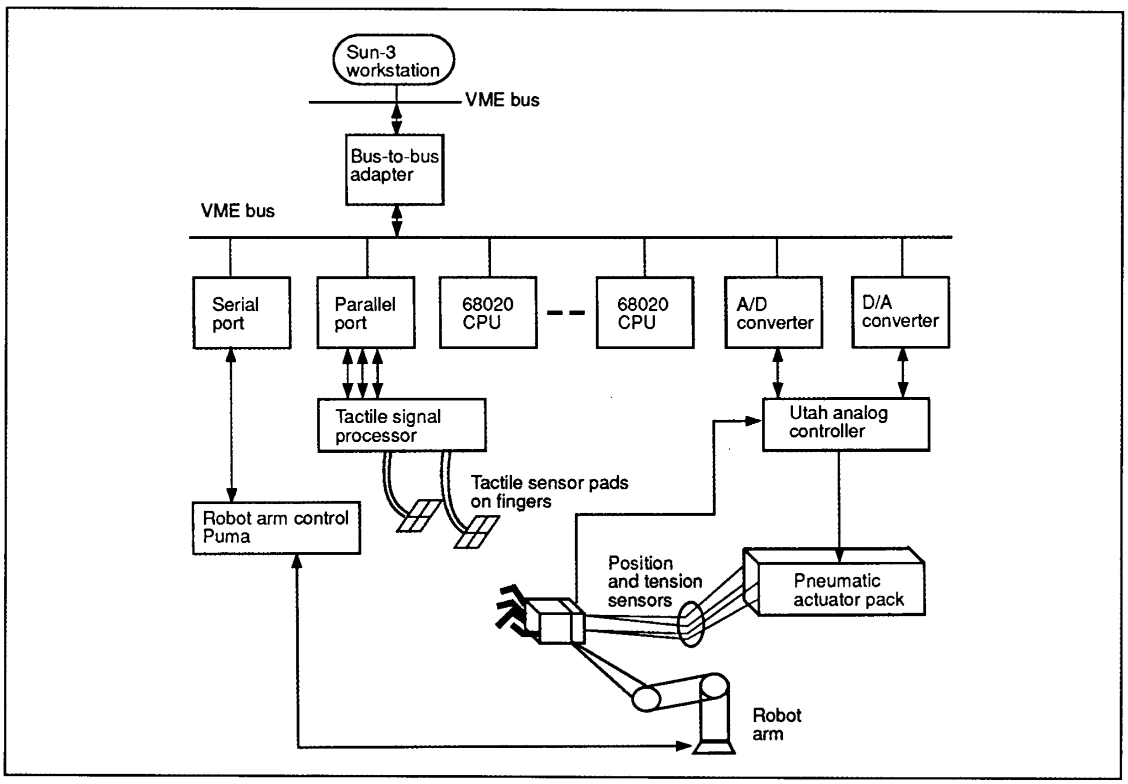

Figure 2. Hardware overview.

task-level scripts, which can then be bound to particular sensors, actuators, and methods for accomplishing a generic grasping or manipulation task. The instruction set Dial supports is specified at runtime in a user-supplied back end.

The task of removing a light bulb from a socket requires tendon-force feedback to determine the grasp strength and the coordination of the fingers moving in parallel to unscrew the bulb. The task is performed by moving the arm to the light bulb while preshaping the hand, and then repeating the following sequence of events: grasp the light bulb, turn the hand/arm counterclockwise, release the grasp, and move back to the initial preshaped position by rotating the hand and arm clockwise. After the bulb becomes loose, the hand grasps the bulb and retracts it from the socket.

A second application is haptic object recognition — that is, understanding shape from touch. It is becoming clear that robotic systems need capabilities similar to the human haptic system in order to perform complex grasping, manipulation, and object-recognition tasks using dextrous hands. Research in psychology has shown the ability of humans to accurately discriminate an object's shape by touch alone.[5] We have attempted to emulate this human ability by recovering the shape of objects through grasping. We have chosen to model objects as superquadrics. These models are derived from a parameterization that allows a wide degree of freedom in modeling objects. The parameter space is continuous and allows a smooth change from a cuboid to a sphere to a cylinder, with more complex shapes derivable with the addition of bending and tapering parameters. These "lumps of clay" are deformable by the usual linear stretching and scaling operations and can be combined using Boolean set operations to create more complex objects.

In addition to possessing most of the desirable modeling criteria discussed above, the models are surprisingly easy to recover from sparse and noisy sensor data. The small number of parameters needed to recover the shape describes a very large class of objects that can be equated with generic or prototypical recognition. Efficient recovery of superellipsoids (a subclass of superquadrics) from vision and range data has already been shown in the work of Pentland,[6] Solina,[7] and Boult.[8]

What makes superquadrics particularly relevant for haptic recognition is the following:

- The models are volumetric in nature, which maps directly into the psychophysical perception processes suggested by grasping by containment.
- The models can be constrained by the volumetric constraint implied by the joint positions on each finger.
- The models can be recovered with sparse amounts of point contact data, since only a limited number of parameters need to be recovered. There are five parameters related to shape and six related to position and orientation in space. Global deformations (tapering, bending) may add a few more.
- In addition to the use of contact points of fingers on a surface, the surface normals from contacts can be used to describe a dual superquadric that has the same analytical properties as the model itself.

In this procedure, the Puma arm moves the hand to a position in which it will close around the object. Then the fingers are closed by position commands until the observed force (estimated by the difference between the tendon forces) exceeds a given threshold, which indicates that the finger is in contact with the object. The joint-angle positions are read, and kinematic models of the hand and the Puma arm are used to convert them to *XYZ* positions in world coordinates. This sparse set of points is then injected into a recovery algorithm developed by Solina[7] that uses a nonlinear estimation technique to recover the superquadric parameters.

We tested this procedure against a database of six objects, including objects that could be modeled as undeformed superquadrics (block, large cylinder, small cylinder) and deformed superquadrics (light bulb, funnel, triangular wedge).[9] Each object's shape was recovered accurately with extremely sparse amounts of data, typically 30-100 points. Note that this is about two orders of magnitude less than typical range data images, which try to recover shape with denser data that, unlike touch sensing, is limited to a viewpoint that only exposes half the object's surfaces to the sensor. We are currently extending this method to include more exploratory procedures for understanding shape from touch. □

Acknowledgments

This work was supported in part by DARPA contract N00039-84-C-0165, and NSF grants DMC-86-05065, DCI-86-08845, CCR-86-2709, and IRI-86-57151. Thanks also to Steven Feiner and Cliff Beshers for bringing Dial to our attention.

References

1. S.C. Jacobsen et al., "Design of the Utah/MIT Dextrous Hand," *Proc. 1986 IEEE Conf. Robotics and Automation*, Computer Society Press, Los Alamitos, Calif., order number 695 (microfiche only), pp. 1,520-1,532.
2. S. Narasimhan, D.M. Siegel, and J.M. Hollerbach, "Condor: A Revised Architecture for Controlling the Utah-MIT Hand," *Proc. 1988 IEEE Conf. Robotics and Automation*, Computer Society Press, Los Alamitos, Calif., 3-vol. set, order number 852, pp. 446-449.
3. *The Force Sensing Resistor: A New Tool in Sensor Technology*, Interlink, Santa Barbara, Calif., 1986.
4. S. Feiner, D. Salesin, and T. Banchoff, "Dial: A Diagrammatic Animation Language," *IEEE Computer Graphics and Applications*, Vol. 2, No. 7, Sept. 1982, pp. 43-54.
5. S. Lederman and R. Klatzky, "Hand Movements: A Window into Haptic Object Recognition," *Cognitive Psychology*, Vol. 19, 1987, pp. 342-368.
6. A.P. Pentland, "Recognition by Parts," Tech. Report 406, SRI Int'l, Menlo Park, Calif., 1986.
7. F. Solina, *Shape Recovery and Segmentation with Deformable Part Models*, PhD dissertation, Dept. of Computer Science, Univ. of Pennsylvania, Philadelphia, Dec. 1987.
8. T. Boult and A. Gross, "Recovery of Superquadrics from Depth Information," *Proc. AAAI Workshop Spatial-Reasoning and Multisensor Integration*, Morgan Kauffman, Los Altos, Calif., Oct. 1987.
9. P. Allen and K.S. Roberts, "Haptic Object Recognition Using a Multifingered Dextrous Hand," Tech. Report CUCS-363-88, Dept. of Computer Science, Columbia Univ., New York, Aug. 1988.