TOWARDS PERCEPTUAL ROBOTICS[1]

Thea Iberall and Damian Lyons

COINS Technical Report 84-17

(August 1984)

Laboratory for Perceptual Robotics
Department of Computer and Information Science
University of Massachusetts
Amherst, Massachusetts 01003, U.S.A.

Presented at the *1984 IEEE International Conference on Systems, Man, and Cybernetics*, Technical University of Nova Scotia, Halifax, Nova Scotia, October 9-12, 1984.

---

[1] Preparation of this paper was supported in part by grant NS14971-65 from NIH and ECS-8108818 from NSF. Professor Arbib has recommended this document for technical report publication.

Abstract

This paper investigates some of the ways in which the study of human anatomy, physiology and behavior can contribute to the design of a versatile robot system. In particular it concentrates on the integration of dynamic sensory information into motor activity. We use schema theory for analyzing perceptual structures and distributed motor control. A computing scheme and control architecture based on this approach is broadly outlined, and then is applied to the control of a dextrous robot hand. We call this approach Perceptual Robotics.

TABLE OF CONTENTS

1.0 INTRODUCTION 5

2.0 INDUSTRIAL ROBOTS 6
- 2.1 Issues in Robotics 6
- 2.2 Limitations of Industrial Robots 7

3.0 SENSOR-BASED ROBOTS 8
- 3.1 Use of Sensors in Robots 8
- 3.2 Limitations of Sensor-Based Robots 9

4.0 WHAT WE CAN LEARN FROM LIVING SYSTEMS 9
- 4.1 Action and Perception 10
- 4.2 Sensations in Humans 10
- 4.3 Perception in Humans 10

5.0 THE HUMAN HAND AS A MODEL FOR A PERCEPTUAL ROBOT HAND 11
- 5.1 Prehensile Movements 11
- 5.2 The Human Hand as a Tool 12
- 5.3 Preshaping the Hand for the Task 13
- 5.4 Power vs Precision Grasping 14

6.0 PERCEPTUAL ROBOTS 16
- 6.1 Schema Theory Integrating Sensation and Motor Control 16
- 6.2 Perceptual Robot Hands 17

7.0 THE LABORATORY FOR PERCEPTUAL ROBOTICS 17
- 7.1 The Problem Domain 17
- 7.2 A Perceptual Programming Paradigm 18
- 7.3 The Grasp Schema 20
- 7.4 Grasp Simulation 23
- 7.5 The LPR Hand Control Network 24

8.0 SUMMARY 24

9.0 ACKNOWLEDGEMENTS 25

10.0 REFERENCES 25

LIST OF FIGURES

Fig. 1: VIRTUAL FINGERS 14

LIST OF TABLES

Table 1: TRANSDUCERS OF SENSORY INFORMATION 8
Table 2: TASK OVERVIEW 15
Table 3: PREHENSILE CLASSIFICATION 21

§1 INTRODUCTION

Designers of industrial robots have found viable, real time solutions to problems in kinematics, dynamics, robot training, and tool gripping, using simple feedback mechanisms to determine position, contact, and velocity. In laboratories, sensors are being developed which provide more complex tactile, motion, and visual information; sophisticated grippers are being developed for general purpose interactions; and languages designed for enhanced human/robot interactions. From another perspective, researchers in the neurosciences have studied sensory signaling and motor control in living systems, looking into issues such as feedback/feedforward mechanisms and distributed control. Mechanoreceptors in the muscles, joints, and skin, contribute to the sense of *kinesthesia*, the sense of position and movement. Sensory information from body stimulations, is processed by the central nervous system (CNS), to become *perception*, the awareness of the body in relation to the environment. Tactile and visual perceptions allow object recognition and movement awareness. A third related area of research is limb prosthetics, where devices ranging from simple muscle controlled hooks to microprocessor-aided limbs are being developed and tested.

We are defining *perceptual robotics* as the area where these fields converge. Advanced sensor-based robots will need both *sensor models* for feedback mechanisms and will need *processing models* for perception. Neuroscience needs models and *testbeds* for understanding experimental data. Both of these fields can contribute to the advance of limb prosthetics, through enhanced artificial limbs and through a better understanding of the central nervous system.

At the Laboratory of Perceptual Robotics (LPR) at the University of Massachusetts, we are initially developing a testbed for one particular problem domain in perceptual robotics, that of a distributed sensory control structure for multiple robotic manipulators. The approach taken at LPR is three fold. Firstly, we are conducting and analyzing *human skills studies* to explore how humans reach for and grasp objects. From these studies, we are constructing classifications for movements and for perception, developing motor programs involving the activation of *schemas*. Our data has indicated that we can take a functional, or *virtual*, approach to defining tasks. We pay close attention to the neurophysiological literature for understanding hierarchical and distributed processing as done in the central nervous system. In this paper, we discuss functional approaches to issues in perceptual robotics that include such topics as multiple degrees of freedom, active sensing, object representation, anticipation, task definitions, and perception.

Secondly, to test our schemas and classifications, we have constructed a *graphic simulation* of the perceptual robot testbed. It was designed in such a way to allow testing of various ideas in visual and tactile sensation and perception. With the simulator, we can compare data from our schema based model to the data gotten experimentally from human subjects. In this way, we can both develop our model and fine tune it.

Finally, we are in the process of building a multiprocessor, hierarchical architecture for control of multiple manipulators in a perceptual robot. We are integrating static and dynamic sensory information captured from cameras, force sensors, and LPR-developed tactile array sensors. This testbed allows the incorporation of image processing programs developed at UMass for feature extraction, optic flow analysis, stereoposis, and object recognition. It also allows the incorporation of LPR programs for tactile perception and kinesthesics using the tactile and force sensors.

§2 INDUSTRIAL ROBOTS

2.1 Issues in Robotics

The robots used in today's factories are robot arms capable of picking up objects and moving them from place to place. Movement in the majority is based on position control [8]. In those cases where a more sophisticated control scheme is used, the robot is still 'taught' in terms of positional information or programmed in a language (e.g. VAL) which has no concept of an object other than as occupying a fixed position.

A robot arm is composed of links and joints, where each joint provides at least one *degree of freedom* (dof). The hand of an industrial robot is called an end effector, and the kinematic issue is to determine the position and orientation of the end effector relative to other joints [34, 39]. A major problem in the control of multiple link robots is the coordination of multiple degrees of freedom.

Besides the kinematic problems, there is the problem of dealing with the forces acting on the robot arm, due to movement as well as to gravity. *Statics* involves the balance of forces without movement, and *dynamics* involves the application of forces to perform movement, and also the compensation for forces which arise during movement (gravity loading, nonlinear coriolis forces, centrifigal forces, and inertial forces).

When a robot arm moves, its overall geometry changes, and thus the static forces acting on it will change. A controller must calculate what these forces will be, and compensate for them. In an industrial situation where time is money, one would want the robot arm to move as fast as possible. But the faster the arm moves, the more complex the force calculations.

Robot sensors provide the information about the environment which is needed to determine such things as location of limbs, shape of hand, contact with objects, and forces currently acting on links. This information can be used to provide feedback for the robot control system. In control theory, a *servo mechanism* is a feedback control system in which the output is some mechanical position, velocity, or acceleration. It is a *closed loop* system, where the set point (the desired state) can be used as a reference state for the system. A negative feedback system measures of the actual state of the system (e.g., its position, its velocity, etc) and then computes the error based on a desired state. In order to reduce the error, the controller will then modify its input to the system. In such a scenario, the control problem for making a robot arm track a desired position trajectory would be to calculate the torques to apply to servo all the joints in real time.

2.2 Limitations of Industrial Robots

Industrial robots based on position control require a well structured work environment before they can operate effectively [27]. Three features characterize these robots: few degrees of freedom, little use of dynamic sensory information, and limited control architectures.

Typical industrial robots have 4-6 degrees of freedom. By ensuring that the robot arm is purposely designed for a particular range of tasks, robot control strategies do not have to deal with the many degrees of freedom which would be needed for a completely versatile robot system. As an example, many robot arms move in a Cartesian coordinate system, which is simplier for positioning than is a revolute joint system. Also, the end effector is constructed more like a hand holding a tool than a hand itself, thus reducing the degrees of freedom. The robot arm ends in a general clamping mechanism, such as an electromagnet or a bayonet socket; the end effector clamps to the arm and is designed with some special task capability, such as the capability to hold objects, open car doors, or tighten a screw. The problem associated with these approaches is that they limit the tasks which a particular robot system can undertake.

Industrial robots also minimize the sensory information needed. This is achieved by enforcing a structured work environment. As a result, industrial robots need a high degree of positioning accurarcy, and repeatibility (i.e., to return to a given point). The advantage is that they can work without visual and tactile senses, but again, the problem with this approach is that it limits the task versatility of a robot system.

Working from the above simplifications ensures that the control architecture will be simple. However, any attempt to gain more versatility (e.g., having more than a few degrees of freedom, increasing the amount of sensory input, etc) will also need a reworked structure for the control architecture. Providing a lot of degrees of freedom, or a lot of potential sensory input is not enough: a sophisticated control architecture is needed to integrate these components.

Industrial robots are good for simple assembly line tasks, and work well in a highly structured environment. Robots of the future will be needed in less structured environments and cannot afford to be limited to a narrow range of tasks.

§3 SENSOR-BASED ROBOTS

3.1 Use of Sensors in Robots

The sensor-based robots, newly introduced to the factory or still being designed in laboratories, have one or more sensors that do more than just determine position of the robot arm. Some of the physical features and the types of transducers being used to measure them are listed in Table 1.

Position sensing of a link in space can be performed using position encoders: a disk containing encoded positions turns when the joint moves, and an optical sensor reads the position from the disk. Transformers and resolvers are also used, which measure voltages as being proportional to the position. Other transducers for position include potentiometers which measure resistance, and ranging devices, such as sonar, radar, or lasers. The electrical signal from the motor can be directly measured as well.

PHYSICAL FEATURE	IN ROBOTS	IN MAMMALS
position	direct measure, encoders, resolvers, potentiometers, ranging devices, cameras, photodetectors	muscle spindles, joint receptors
proximity	photodetector, camera	hair cells
contact	proximity detectors, binary switches	Pacinian and Meissner cells
pressure	optical fibers, conductive polymers	Merkel cells, Ruffini endings
force	strain gauge, direct measure	Golgi tendon
velocity	tachometer, diff. transformer, direct measure, camera	muscle spindles, joint receptors
acceleration	accelerometer, camera	otoliths, semicircular canals
simple object features (location, movement, etc)	above devices, cameras, photosensitive elements	retina, spindles, cutaneous receptors

Table 1: TRANSDUCERS OF SENSORY INFORMATION

The proximity of an object is sensed using infrared devices or photodetectors. Actual contact can be determined by using proximity detectors, or simple binary switches.

Pressure or tactile sensing can be done using conductive polymers, by measuring their resistance under deformation; or optically, by measuring the light intensity as being proportional to the force applied against the surface. Force sensing orthogonal to the links is done using strain gauges along links and in a wrist. A strain gauge measures resistance proportional to cable tensions. These forces can also be measured directly from the motor based on its back EMF. To avoid excessive force calculations while inserting an object into a hole, passive compliance is used, where the object is jiggled by the end effector until the force against it reduces.

Velocity sensing is done using a tachometer or a voltage transformer, which measures differentially induced voltages. Acceleration can be sensed using an accelerometer to measure the force and then compute the acceleration using a small given mass.

The sensing of the environment, and objects in it, can be done with many of the above devices. Knowing the position of the end effector, and making contact with an object tells the location of the object. Relative movement between object and robot arm is similar. With a camera, visual sensing is possible. Static vision provides position information, while dynamic vision allows movement determination. The object's direction of movement in the field of vision is determined by object flow fields, vectors which expand from one point (the focus of expansion) as an image changes over time [5, 10]. Involved processing is needed to determine more complex object and environment features, such as textures, object shapes, and object recognition.

3.2 Limitations of Sensor-Based Robots

Sensor-based robots will be required to perform complex tasks as repair and one of a kind tasks; i.e., to repair machines, and to deal with new objects. With their sensors, they can work in a less structured environment than the industrial robots. In order to interact with objects, robots must have a representation of those objects. Under development are various world-modeling robot systems, which present the user with a looser tie between object and position [1, 24, 26]. However, we maintain that the lack of sufficient sensory information is not the only problem, but also the *interfacing* of sensory information to motor behavior. Adding one or more of the above sensors does not automatically increase the robot's versatility. This paper suggests an approach to a high-level control architecture which is oriented at this more versatile robot system.

§4 WHAT WE CAN LEARN FROM LIVING SYSTEMS

4.1 Action and Perception

In comparison to today's robots, even the simplest animals have more degrees of freedom and a more robust control architecture. Besides having the CNS as a highly integrated processing apparatus, animals such as mammals have evolved sophisticated sensors in their periphery. Mechanical and chemical receptors provide sensory information that contribute to perceptions of the environment in relationship to the organism. In the *action/perception cycle* [2, 32], an organism performs actions guided by expectations. The organism perceives the consequences of these actions on the environment, and therefore actions both confirm assumptions and provide new information about the environment.

4.2 Sensations in Humans

Within the muscles of humans, static muscle spindles signal the stretching of the muscle, thus indicating the current position of the end of the limb. Their highest density is found in the muscles needed for fine movement (hand, foot, neck) and for posture (soleus) [37]. Receptors at the joints signal joint angles, again contributing to limb position. *Proprioception*, or the reception of stimuli within the organism, is the feedback from these types of internal sensors to the CNS.

At the skin, hair cells can detect proximity to objects. Pacinian and Meissner cells in the skin signal contact. It is thought that Merckel discs and Ruffini endings sense pressure applied to the skin, by signaling using frequency encoding [37]. Other receptors within the tendons, called Golgi tendon organs, are sensitive to force, and work in a way similar to strain gauges. Dynamic spindles in the muscles signal the velocity at which the muscle length is changing, and in the joint, receptors signal the changing angle. The otoliths and semicircular canals within the inner ear act as an accelerometer in the head, contributing to the overall sensation of acceleration of the body.

Pacinian and Meissner cells, besides signaling contact, also signal slow and fast vibrations against the skin. As a result, they provide relative movement information between an object and the skin. These too are in highest density in the sensitive skin areas (fingers, tongue, lips, etc).

The proprioceptive and cutaneous sensors all convey information about the *somatosensory* state of the organism to the CNS. Non-tactile sensory information can be obtained through receptors in the retina of the eye. In the CNS, further processing is performed in order to perceive complex object and environmental features.

4.3 Perception in Humans

Spatial perceptions, such as two-point discrimination, depend on the resolution and location of cutaneous tactile sensors. Temporal tactile perceptions also exist, which involve relative movement between the skin and an object [11]. Stereognosis, the perception of three dimensional structure (object size, shape, and orientation), involves knowing which movements are being performed [20]. If different parts of an object are brought into contact with a blindfolded person's outstretched palm, the object cannot be identified [11]. The person is not performing the manipulations, and thus cannot create a mental image of that object over time. This is *active touch*.

The kinesthetic sense of position and movement is a complex perception using not only the mechanoreceptors in the muscles, tendons, joints, and skin, but the chemical receptors in the retina as well [23, 29]. Involved in kinesthesia are also the perceptions of effort, force, and weight [37]. Non-tactile perceptions of objects, beginning as sensations in the retina, lead to perceptions of environmental and object features [15, 37] and also the awareness of movement, or dynamic vision [10].

§5 THE HUMAN HAND AS A MODEL FOR A PERCEPTUAL ROBOT HAND

5.1 Prehensile Movements

The action/perception cycle is seen in the way humans use their hands in *prehensile*, or grasping, movements. An object is perceived through the visual system. As the person reaches for the object, the hand preshapes in anticipation, using proprioceptive feedback, into a shape suitable for the interaction. The action eventually produces stimuli against the skin as the hand touches the object; these stimuli modify further actions as the hand encloses around the object. As a model for a dextrous perceptual robot hand, the human hand provides a rich source, being a complex organ capable of both performatory and perceptual functions [36]. The understanding of human prehension seems crucial to the development of perceptual robots, as their versatility will be directly affected by their ability to interact with objects.

Grasping is one component of reaching movements [7, 17, 18]. It involves wrist movements for hand orientation, and finger and thumb movements for preshaping, gripping, and manipulating the object, involving in all over 24 degrees of freedom. The other component, the spatial one, involves arm movements coordinating three degrees of freedom at the shoulder and one at the elbow, bringing in such issues as target and depth definitions.

The behavioral studies of Jeannerod [16, 17, 18] suggest ways of looking at hierarchical strategies for the coordination of the degrees of freedom. Jeannerod describes the spatial component as a two phase arm movement: first, a (fast) *ballistic* move to a location near the object, and then, a (slower) *tracking* move to the object. During these movements, the grasping component is active. In a *preshaping* movement, the fingers extend in preparation to grasp the object, and then, in a *gripping* movement, the fingers flex to grasp the object. Their evidence suggests that the timing of the preshaping relates to the ballistic move, and the timing of the gripping relates to the tracking move [17].

Evidence from ablation studies indicates that separate and parallel pathways are involved in these two components of reaching. The control of fine finger movements is permanently impaired in monkeys, apes, and man after a pyramidotomy (interruption of the pyramidal tract) or ablation of the motor cortex [7]. From these types of studies, we feel justified in analyzing hand movements separately from limb movements.

5.2 The Human Hand as a Tool

The human hand is a dextrous manipulator; it is a general purpose tool that can be shaped into "the necessary tool for the task." It can take on a variety of shapes which, in effect, limit the number of available degrees of freedom to match those needed for the task. Not only is a new special purpose tool created in hand preshaping (the kinematic issue), but the tool is created in such a way as to optimize the use of the moments of force (the dynamics issue). As humans decide to grasp objects, they make decisions on how to grasp them. In a task as simple as picking up a pen in order to write with it, a person might use various strategies. When the pen point is facing away from the body, a person might pick it up using the same grip he or she will be writing with. This uses extra effort to twist the wrist and arm, but minimizes the effort used to reorient the pen after it is gripped. The person might instead take it as it is, and then reorient it in his or her fingers. This minimizes the stress on the wrist and arm, but uses extra effort to reorient the object. The chosen strategy is a matter of what minimizes the person's own sense of effort.

The study of the evolution of the human hand shows the development of an independent index finger, highly dependent ring and little fingers, and a saddle joint at the base of the thumb, which gives humans the ability to bring the thumb in opposition to the rest of the fingers. The basic architecture of the hand provides "functions for free" at the lowest level of a control hierarchy: the fingers, when being flexed, will automatically move inward (adduct); when extended, they will move outward (abduct). This simple example shows the interactions of the many degrees of freedom, defined by Tubiana as *kinetic chains* [38].

These types of limitations in the hand are further seen by analyzing typical prehensile tasks, and by looking at the relationships between the bones, tendons, and muscles. For a stable grip, the wrist must be stable, since if the wrist moves, the lengths of the long finger flexors change and the applied forces will vary. In gripping an object, it is also important to maximize contact (within a task context, of course), in order to maximize somatosensory knowledge about the object and, therefore, stability. In this regard, the architecture of the hand forces a cylindrical object (such as a hammer or a screwdriver) to be gripped obliquely across the palm, maximizing the skin surface contacting it. For holding small objects in the fingertips, a very slight second degree of freedom at the distal joint (the joint furthest from the body) on the thumb and at the middle joint on the fingers brings the fingerpads toward the thumb pad, thus again maximizing contact [21, 38].

5.3 Preshaping the Hand for the Task

Preshaping the hand can be done under somatosensory guidance and/or visual guidance. In experiments where subjects could not see their hands while reaching for various objects, Jeannerod showed that preshaping can be done under somatosensory guidance alone [16, 17]. The shape that the hand took on was the same shape with or without visual guidance. Whereas, in studies on a patient with a parietal cortex lesion [19], Jeannerod shows that the woman, who was without somatosensory feedback, could not preshape her hand unless it was within sight. Although it was sufficient for gross movement, she could not duplicate the dexterity of a normal person.

The preshaping of the hand has been shown to take on a functional, or *virtual*, characteristic when attempting to grasp objects of different sizes [3]. In a task such as reaching to grasp a mug by its handle, we found that by varying the size of the mug handle, the shape which the hand takes on varies as well. As seen in Figure 1, if the object is a teacup with a small handle, the hand approaches it with only the index finger extended. If the handle is a bit larger, two fingers are extended; if the mug is a beer mug, three or four fingers are extended. The suggested mechanism behind this is a *virtual finger* representation, as a hierarchical substructure in hand control. This task would be represented in the higher levels of the CNS as a three fingered task, since three basic forces must be brought to bear in this task: inserting a virtual finger into the handle, bringing a virtual finger to a place on top of the handle, and (perhaps) supporting the handle from below. The number of real fingers mapped into a virtual finger does not affect the task definition. Also, a task such as this does not need multiple representations at higher levels of the CNS. A task can be defined in terms of the number of virtual fingers needed to conceptualize the bringing to bear of the necessary forces, while the actual playing out of the movements during the task is done at a lower level across the real fingers and hand.

Hollerbach [14] used a similar approach in analyzing the task of handwriting, finding four key issues in manipulating the pen. These include maintaining pressure on the pen, maintaining pressure against the paper, providing movement from left to right (in writing English), and providing movement up and down to form the letters. These define the key parameters needed for the task definition, or the *functional degrees of freedom*.

The study of preshaping movements gives clues to the grasping process, as they are those which restrict the available degrees of freedom to the ones needed in the final task. Concepts such as virtual fingers and functional dofs provide an analysis approach to understanding human prehensile movements.

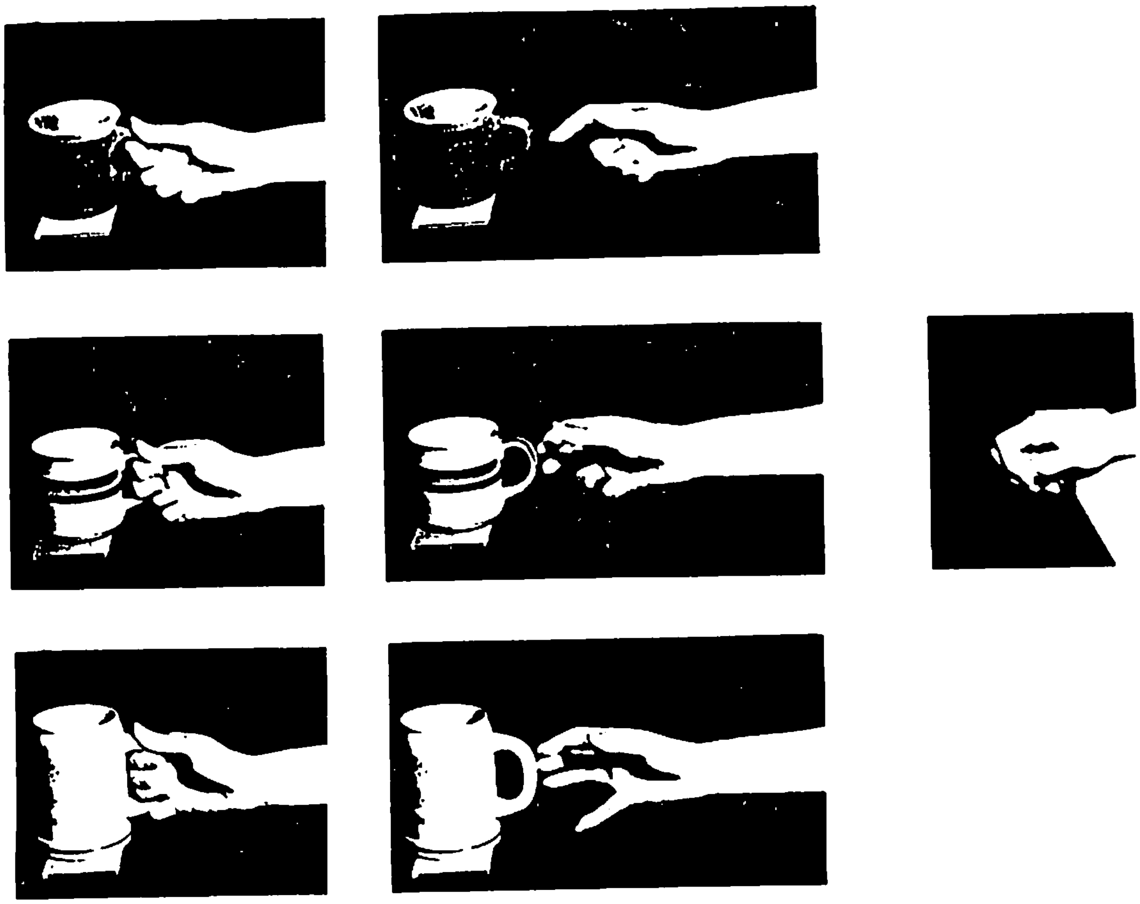

Fig. 1: VIRTUAL FINGERS
Different mappings for virtual fingers for different size objects (from [3])

5.4 Power vs Precision Grasping

A hand shape taxonomy has been defined in the literature for prehensile movements [31, 22, 25, 28]. Napier described a *power grip* (a grip using the whole hand), and a *precision grip* (a grip using the tips or sides of the fingers). Napier [30, 31] points out that a prerequisite for prehension is the stability of the object, which depends on the orientation of the thumb. Within this classification, a variety of grasping characteristics and subdivisions have been defined [22, 25]. These include *precision handling*, involving rotations and translations of objects held at the fingertips, and power grips such as the *screwdriver squeeze*, the *hammer squeeze*, the *spherical squeeze*, etc. *Pinch*, between the thumb and first finger, is a subdivision for the exertion of force for manipulating smaller objects. A list of tasks based on Long's subdivisions [25] is seen in Table 2.

TASK	FEATURES OF TASK	GRASP
driving a screw with a screwdriver	**preshape:** power **thumb:** abducted **wrist dof:** twist	screwdriver squeeze
driving a nail with a hammer	**preshape:** power **thumb:** adduct or abduct **wrist dof:** side to side use elbow, shoulder	hammer squeeze
lifting a cylinder to carry it	**preshape:** power **thumb:** adducted	simple squeeze
twisting jarlid open	**preshape:** power **thumb:** abducted **wrist dof:** side to side	disc squeeze
lifting ball to throw	**preshape:** power **thumb:** adducted **wrist dof:** up and down	spherical squeeze
lifting suitcase to carry	**preshape:** power **thumb:** abducted use elbow, shoulder	hook grip
lifting mug by handle for drinking	**preshape:** power or precsn **thumb:** abducted **wrist dof:** twist use elbow, shoulder	simp. squeeze or pinch
turning bolt with fingers	**preshape:** precision **fingers dof:** fl/ex/add/abd **wrist dof:** twist	precision rotation
writing with pen pulling on spring	**preshape:** precision **fingers dof:** fl/ex/add/abd	precision translation

Table 2: TASK OVERVIEW

In the precision grip, the muscles act isotonically and dynamically, with the fingers flexed, the wrist slightly extended, and the thumb in opposition with the finger pads. The radial fingers (thumb, index and middle) form a "dynamic tripod" [38], and are generally less flexed than the ulnar fingers (ring and little). Directly opposing the thumb, the index finger can have less flexion. The thumb itself has a great deal of mobility due to its saddle joint and its five degrees of freedom [38].

The power grip is isometric and static. The wrist becomes slightly extended, deviating toward the ulnar side, and the thumb aligns longitudinally with forearm [31]. For the power grip, an abducted thumb provides some precision; an adducted thumb provides a stronger stabilizing influence. The intrinsic muscles (those within the hand) rotate the fingers to accommodate the shape of the object, optimizing contact between the finger pads and the object, and allowing the extrinsic flexors (those inserted into the forearm) to do the major part of the gripping. The ulnar fingers converge on the base of the thumb, tending to force the grip into an oblique palmar one, instead of a transverse palmar grip. Even the little finger, although having the least strength in flexion, has a special role in the power grip, due to its peripheral position, its mobility, and capacity to press an object against the palm.

In its unique position, the middle finger works well in both the power and precision grip: it can align with the ulnar digits for a power grip, or with the radial digits for a precision grip [38].

§6 PERCEPTUAL ROBOTS

6.1 Schema Theory Integrating Sensation and Motor Control

A *perceptual robot* not only has multiple sensory windows on its environment, but it also *integrates* sensory information into goal-oriented motor behavior. Sensors measure external environmental features. The stimulations of the sensors provide *sensations*, which are integrated and processed into *perception*, which could be described as the sensory state of the robot *within* a task context. Sensations may have different interpretations for different tasks, and to some tasks they may be irrelevant. Thus perception is intimately tied to the current task, and is integrated completely with it. We argue that this provides an easier and more robust way to program robots.

Living systems have evolved successful strategies for movement in a complex environment, and robotic systems have brought to light what the major issues are in movement control. By sharing concepts from the neurosciences with those in robotics, researchers in both areas can learn from each other: Perceptual robots need both models for sensory feedback mechanisms and need processing models for perception; neuroscientists need models and testbeds for understanding their experimental data. In the long run, both of these fields can contribute to the advance of such areas as limb prosthetics, where the integration of sensory processing and motor control is crucial.

Schema theory provides such a mechanism, linking robotics with the neurosciences, as it can be used to model a possible way the CNS organizes complex activities [2]. A schema monitors feedback from the system it controls in order to tune its activities, and provides a unit of motor control within an overall coordinated control program [3]. Cooperating and competing schemas each monitor an aspect of an activity, and the overall behavior of the system is the combined behavior of all of its component schemas.

6.2 Perceptual Robot Hands

We would like to unify concepts in the robotics literature with behavioral studies on human grasping and reaching. In this regard, we define the *Grasp schema* to reflect the steps used by humans in grasping. Specifically, these steps are: the grasp preparation (the *preshape*), the gripping of objects (the *grip*), and the manipulation. This unifies what has been called static grasping (the grip) with dynamic grasping (the manipulation) [9]. A *perceptual robot hand*, a dextrous hand based on a model which resembles the human hand, would have the ability to shape into the necessary tool for a given task, using tactile and visual perception. It could be used for direct manipulation of small objects, rotating, rolling and sliding them to desired positions and orientations. It could be attached to a limb (robot or someday even human) to allow reaching as well.

Advantages of a perceptual robot hand over a standard robot gripper include the ability to stably grip a wider range of objects, directly manipulate held objects, and the ability to be used as a versatile sensory exploration tool. Such a dextrous hand provides a much greater grasping capability than a two fingered gripper. In addition, any object can usually be grasped in more than one way, thus allowing task issues other than object shape to be considered. With more degrees of freedom in the fingers, it can also manipulate the object once it has been grasped. Tasks, many times, involve a variety of actions to be performed. Finally, since it can be dynamically reshaped, it would be a better exploration tool than the limited industrial end effector.

§7 THE LABORATORY FOR PERCEPTUAL ROBOTICS

7.1 The Problem Domain

At the Laboratory for Perceptual Robotics (LPR), we are currently exploring one particular problem domain: that of the real-time control of a perceptual robot limb equipped with a dextrous hand.

The dextrous hand which the LPR is currently enhancing perceptually is the Salisbury hand [35]. The hand is an articulated three fingered hand with 9 degrees of freedom, and it is currently being added to a 4 degree of freedom Cartesian robot arm. It is being provided with sensory devices such as CID cameras, force sensors, and tactile sensors. Current work is going on to integrate static and dynamic sensory information captured from the sensors, and to incorporate image processing programs developed at UMass for feature extraction, optic flow analysis, stereoposis, and object recognition [5, 12].

The UMass VISIONS system [12] uses a schema architecture involving a low level system for image segmentation from local information, and a high level system which knows about various world objects. The high level system provides a global context for further local segmentation, and cooperative computations allow hypotheses to be generated and discarded until the system converges on a good interpretation of the image [5].

To determine the movement of an image in the field of vision, dynamic vision [10, 5] is used. The direction of the movement is determined by object depth, using the focus of expansion of a vector field of displacements over time.

At the LPR a variety of tactile sensors have been built [6, 33]. The Overton tactile array sensor [33] uses conductive polymers to measure resistance as proportional to force applied to the surface. The Begej optical tactile sensor [6] measures the intensity of the light reflected at a given location as proportional to force being applied there. These are similar to mechanoreceptors in the human skin which sense the intensity of pressure applied.

A schema based controller for the integration of motor commands with sensory and perceptual processing is being built. It involves a multiprocessor, hierarchical architecture, which will be further described later. We would first like to describe how schemas might interact in order to implement our perceptual paradigm, using some simple examples.

7.2 A Perceptual Programming Paradigm

Schema theory is a distributed control concept similar in some ways to Actors [13]. The set of all schemas defined in a particular robot system is similar to long-term memory in a human, or a program library in a computer. When schemas are *instantiated*, they are provided with both specific parameters and a context. Not executing until it is instantiated, a schema can be instantiated as many times as desired. Different instantiations of the same schema are differentiated by their *instantiation number*. An important property of schemas is that instantiations can be grouped in tightly-coupled communication networks called *schema assemblages*; such an assemblage can in turn be treated as a single schema instantiation. This assemblage provides instantiations with a context: it is possible for the same instantiation to behave completely differently when created within different assemblages.

Direct sensory data is hidden from the system by schema instantiations which accumulate the data continuously and present it in a *task-oriented* form. In the domain of obstacle avoidance, the important domain details of sensed objects are mainly geometric, and an instantiation of the *Obstacle schema* will provide this information to the obstacle avoidance task. The schema could do this by culling the information from one sensor or from many, or else perhaps by computing its information from internal representations. The information provided is updated constantly to represent the current world state. In a similar fashion, a *Limb schema* instantiation may produce the position of the limb's endpoint and velocity, although the details of how these were produced is hidden. Instead of "frame-oriented" sensory primitives, the emphasis is placed on the task structure; how the robot manipulates objects, rather than how it recognizes objects.

Sensory information is thus thoroughly integrated into the task in which it is used. Different tasks may call for different views of the same real-world entity; for example, several different versions of the *Limb schema* may be concurrently used for different tasks. Since they all refer to the same real world object, we group these instantiations into an assemblage. Whatever data is common in the assemblage is filtered through component instantiations to the tasks. An instantiation can use information derived by other instantiations in the assemblage, as well as cull new sensory information. It can also compute, by communication with instantiations not in the assemblage, some new features. Most commmunication, however, occurs between instantiations within the assemblage. The assemblage for a particular object is, in effect, the total *internal model* of that object. However, unlike a more standard approach to modeling, multiple windows on the object (each a particular task-oriented view) are present. This representation clearly facilitates a hierarchical object representation, but does not force one.

In a similar fashion, a particular task is represented as a schema assemblage. The task is broken into modular components, some of which will run in parallel and some of which run serially. To illustrate these concepts, we provide two simple examples.

Simple Obstacle Avoidance Example. We would maintain that the most basic property of any real-world object is its ability to act as an obstacle. Therefore, for any schema assemblage representing an object, there will be at least one instantiation of the *Obstacle schema*. Of course, other instantiations may identify other properties of the object (e.g. its use as a tool, or as part of the manipulator).

1. Let there be one instantiation of a *Limb schema* for each limb in the manipulator.
2. For each real-world object there is one instantiation of the *Obstacle schema*.
3. Each *Obstacle schema* instantiation outputs the position of the centroid of the object, the radius of a sphere tightly enclosing the object, and the radius of a 'care' sphere around the object.
4. Each *Limb schema* will compare its position with the 'care' sphere around nearby objects, and if its position comes within the sphere, it will cause the limb to have a velocity component away from the object, proportional to its distance along the radius of the 'care' sphere.

The instantiation mechanism allows the above algorithm to be easily expandable in the number of obstacles and manipulator limbs. It assumes a concurrent environment to alleviate the resulting time complexity. Also, the above task can run independent of other tasks allowing a series of moves to be programmed without worrying about obstacle avoidance. Of course the naivety of the algorithm may give rise to problems; however, the example is to only illustrate the flavor of schema programming.

Triggering a Task to a Perceptual Stimulus Example. An assembly subtask may perhaps trigger when a principal component has been identified, or a corrective action taken if a non-standard event should happen during the assembly (such as a component being defective, or some foreign body entering the workspace). Again we wish to show the flavor of the style of computation, rather than a particular algorithm.

Each task has a schema whose duty it is to recognize the main object components of the task and instantiate task-appropriate schemas into the assemblages for these objects. Once this has been done, the task can be triggered using the output of these schema instantiations. We have stated that an *Obstacle schema* is the most general case of a real-world object. Let us consider triggering a mug grasping task [3]:

1. A *GraspWatch schema* is instantiated. It looks at instantiations of the *Obstacle schema* for hints that the obstacle may be a mug. If such clues are present, a *Mug schema* is instantiated. The *GraspMug schema* is also triggered with a pointer to the instantiated *Mug schema*.
2. A *Mug schema* instantiation attempts to fit this object as a mug. The relative success of this operation is indicated by the *activation level* of the instantiation. This activation level is essentially the credibility of the hypothesis that this object is a mug. Instantiations with activation levels below a certain threshold just die away. When they die, so do any tasks attached to them. If the activation level is super-threshold, then the *GraspMug schema* attached also has this activation level.

Obviously, if multiple perceptual stimuli surround the robot, many tasks could be triggered at once and begin competing for perceptual clues. Initially, we are not addressing this competition issue but, as suggested in [2], the use of activation levels will allow us to eventually explore the full cooperation and competition paradigm among schemas.

7.3 The Grasp Schema

A *Grasp schema* involves a preshape configuration, a grip configuration, and a set of associated degrees of freedom (typically much less than the sum of all the degrees of freedom in the hand). A perceptual robot hand is equipped with a versatile set of grasps, capable of performing the types of tasks previously listed in Table 1. A grasp is chosen on the basis of three main environmental features: the function which the object is to be used for, intrinsic object features (shape, weight, size), and the environment surrounding the object. Once chosen, the preshape is used to form the hand for anticipation of gripping the object. Once gripped, the object can be moved according to any of the functional degrees of freedom associated with the grasp, which could be task oriented (e.g. rotating a screwdriver) or oriented toward obstacle avoidance (the gripped object can, to some extent, be moved out of the way of obstacles using just the hand; this may occasionally be superior to moving the whole arm). We postulate the intended object usage as the primary index for grasps. The stereotyped configuration produced by this index is then particularised using the intrinisic object properties. Finally the existence of confining obstacles around the object is used to modify the grasp.

The Preshape schema. In order to guide the hand, a *Preshape schema* needs sensory feedback provided by the visual system or the somatosensory system. The visual system provides parameters for the preshape movements, while the somatosensory system informs and verifies that the actual movement is occuring. The *Preshape schema* will put the hand into a particular shape which will allow grasping of an object for a given task. As seen in Table 3, an initial prehensile classification has been developed from our human skill studies. Six basic grasps are defined, grouping the five real fingers as they are used functionally. Going beyond the simple precision versus power analysis, the grasps are based on the amounts of precision and power needed within tasks. It divides the human hand up into its precision and power sides, which is supported by the evidence presented in Section 5. For the three fingered Salisbury hand, the shapes overlap, degenerating into two shapes, basically shape I and V. From this prehensile classification it is possible to list of parameters that the *Preshape schema* needs. These include:

HAND SHAPES	PRECISION			POWER		CHARACTERIZATION	
	VF1	VF2	VF3	VF2	VF3	PREC.	POWER
I Basic Power	1			2345		none	100
II Modified Power	1			234	5	none	100, but hdl size
III Mod. Precisn/Powr	1	2		34	(5)	0-20	80, but hdl size
IV Basic Precisn/Powr	1	2			(345)	50-60	0-50
V Basic Precision	1	2	3		(45)	80-100	0-20
VI Fortified Prec/Pwr	1	23			(45)	60	0-40

Note: () means used if needed; otherwise tucked in or balancing out

Hand shapes represent the six basic prehensile shapes, based on mapping real fingers (1 is thumb, etc) into virtual fingers (VFn). The precision and power characterization was determined using an arbitrary numeric scale (0-100), where a higher number reflects a more intense requirement, and a range reflects dynamic need. Shapes II and III are seen used when the handle size is too small.

Table 3: PREHENSILE CLASSIFICATION

1. *Degree_Of_Precision:* a range, which will affects the choice of the shape, and which is based on the environmental features listed above
2. *Degree_Of_Power:* a range is again needed, based on the same considerations
3. *Object_Size:* a range of objects, from small to large, which affects the size of the virtual fingers; i.e., the degree of adduction/abduction and extension/flexion needed within the virtual finger
4. *Object_Type:* specifically, the shape of the handle or pseudo-handle where the object will be grasped. Many objects have real handles (mugs, screwdrivers, hammers, bolt), and others have an area that can act as a pseudo-handle (cylinder, jarlid, ball, pen)
5. *Functional_Deg_Of_Freedom:* degrees of freedom which are needed during the actual task (such as twist, push, pull, swing around, swing down, put down, etc)

Preshaping for the task is performed by two schemas, the *Precision preshape schema* and the *Power preshape schema*, activated in parallel. The actual behavior depends on the strategies used by these two schemas, which will compete for hand resources (e.g. they will both want control of the thumb, they may both want control of the index, etc), yet cooperate to maintain the hand in a useful posture.

The Grip Schema. The *Grip schema* becomes active when the hand begins closing in on the object. The first part of it involves the anticipation of contact with the object, and the second part involves contour following. The context of the movement is still within the virtual fingers defined by the preshape. The parameters that it needs include:

1. *Object_Size:* again, a range of object sizes
2. *Object_Type:* again, the shape of the handle or pseudo-handle where the object will be grasped
3. *Functional_Deg_Of_Freedom:* degrees of freedom active during the actual task
4. *Expected_Contact_Location:* location of anticipated contact on virtual fingers

The regulation of forces of a grip is important, varying due to weight, fragility, surface characteristics, and utilization. Continued sensory information is needed for compliance and maintaining contact.

The Grasp Schema as a Language Primitive. We divide the robot architecture into a programming level and a real-time control level. This programming level will treat the grasping of an object, its manipulation, and releasing, as *atomic* operations. It sees therefore, not the total hand, but merely the degrees of freedom associated with each grasp; the details of grasping and object manipulation are carried out at the real-time control level. The real-time control will move the gripped object to avoid any obstacles where necessary, using just the degrees of freedom of the grasp.

Examples include: producing the sequence of motor commands to actually grip an object, converting the functional degrees of freedom of a grasp to movement of the hand, or modifying a planned grasp to suit grasping an object in a confined location.

A perceptual robotics language would describe movements in functional and task related, human-like terms, instead of in typical input/output primitives seen in most programming languages. It would avoid the use of frame-oriented sensory primitives, and not force the programmer to worry about the specific degrees of freedom of each joint.

We can go back to the task overview in Table 1 and redefine the tasks. It becomes clear from the architectural analysis of the human hand that the precision/power dichotomy has to reflect the instrinsic object features besides the object type and use. While the table shows using a screwdriver is a power grasp (i.e., shape I in Table 2), with the wrist twisting during the movement, this would not be the case if the screwdriver were small. In that case, it would really be a precision grasp of shape V, where the twisting could in fact be done by finger rotation. For the driving of a nail using a swinging down functional dof, a large hammer would be held in a power grasp of shape I doing the swinging at the elbow; a small hammer would use shape VI using the fingers. Thus, by supplying parameters to the *Grasp schema* such as object size, object type, and the degrees of precision and power needed along with the functional degrees of freedom, the schema determines which hand shape is needed and which real degrees of freedom to use.

7.4 Grasp Simulation

To test various ideas in visual and tactile sensation and perception, we have constructed a *graphic simulation* of a generalised hand: specific models, such as a human hand or a Salisbury hand are inserted as desired [4]. The simulation allows the parallel interaction of schemas to control a graphically represented hand interacting with its environment. It allows the testing of various schema control programs, by providing an interpreter of schemas, and by using abstract representations for hands and objects within the environment. It is allowing us to develop a schema-based robot language.

From the basic simulation, we are constructing a *distributed programming environment*. This environment centers around the schema interpreter. Output from the interpreter can be sent to a graphics display module, to a more limited display or analysis module, or directly to a robot system. In addition, sensory input can be taken from a robot system or from a world simulation module. This simulation aids us in our theoretical development of a schema formalism as a foundation for a robot programming language. It is also useful for the comparison of schema modeled movement with human movement. Currently the system is being written in Pascal on a DEC VAX-11/780; however, we are currently expanding some pilot implementations of this environment on a Symbolics Lisp Machine.

To develop models for grasping movements, we are conducting *human skills studies* to analyze how humans reach for and grasp objects. From these studies further refinements will be made to our prehensile classification, and motor programs developed involving the activation of schemas. Using the simulation in conjuction with these human skills studies, we can compare data from our schema based model to data gotten experimentally from human subjects. In this way, we can both develop our model and fine tune it. We can also transfer our control concepts between robot and human models easily.

7.5 The LPR Hand Control Network

Control of the Salisbury hand, equipped with various sensors, by a single processor is virtually impossible because of the real-time demands. Therefore, we have developed a network topology to control the hand. The topology places one processor at each joint, and over the joint processors of each finger is another processor. The highest level in the hierarchy is the processor directly above the three finger processors. The network topology can be divided into two logical blocks of processors. The hand and three finger level processors can be considered a high-level control unit. Each finger processor and its associated motor processors can be considered a low-level control unit. The processors being used for network, in its current state of being built and tested, are DEC T-11 microprocessors and PDP-11 minicomputers.

The low level unit of the control network is responsible for trajectory generation and servo-control (initially, we use position control, but we are currently working on stiffness control). In terms of control theory, a finger processor is responsible for kinematic transformations and for changing joint torques to motor torques. The finger processors also have a high level function, which is the responsibility for finger coordination; this is achieved through their direct communication with the other two finger processors, and by their access to tactile inputs.

The high-level unit, representing the hardware analogue of a schema assemblage, is responsible for the implementation of the grasp schemas. The input into this highest level is in the form of high-level manipulation instructions, for example Grasp(object) or Orient(object, with orientation), or Roll(object). The exact language for these instructions is being generated from our hand skill studies. The hand processor handles mapping to virtual fingers [3] and grasp configurations.

§8 SUMMARY

We have investigated the characteristics of current industrial robots and found them lacking in a versatility possessed by many of even the simpler animals. We have attempted to apply some observations based on behavioral and neurophysiological research to robotics to generate a more versatile robot system, we call our approach

Perceptual Robotics. We outlined some brief examples to give the flavor of the approach, and then attempted to apply it to the problem of controlling a perceptual robot hand. We have implemented an initial version of the work described here [4], but are still involved in investigating and implementing many of the topics.

§9 ACKNOWLEDGEMENTS

We would like to thank Dr. David McDonald for his helpful suggestions, and our coworkers at the Laboratory for Perceptual Robotics, including Steve Begej, Randy Ellis, Judy Franklin, Darlene Friedman, Gerry Pocock, Nancy Stewart, and T.V. Subramaniyan. We would like to especially thank Dr. Michael Arbib for his continued support, encouragement, and insights into our research.

§10 REFERENCES

[1] A. P. Ambler, I. M. Bellos, and R. J. Popplestone, "An Interpreter for a Language for Describing Assemblies," *Artificial Intelligence*, vol. 14, pp. 79-107, 1980.

[2] M. A. Arbib, "Perceptual Structures and Distributed Motor Control," in V.B. Brooks (Ed.), *Handbook of Physiology--The Nervous System, II. Motor Control*, Bethesda, MD: American Physiological Society, 1981, pp. 1449-1480.

[3] M. A. Arbib, T. Iberall and D. Lyons, "Coordinated Control Programs for Movements of the Hand", COINS Technical Report 83-25, University of Massachusetts, Department of Computer and Information Science, Amherst, Mass., to be published in *Exp Brain Res Suppl*.

[4] M. A. Arbib, D. Lyons and T. Iberall, "A Schema-Based System for Hand Movement", COINS Technical Report, University of Massachusetts, Department of Computer and Information Science, Amherst, Mass., in preparation.

[5] M. A. Arbib, K. J. Overton, and D. T. Lawton, "Perceptual Systems for Robots," *Interdisciplinary Sciences Reviews*, vol. 9, no. 1, pp. 31-46, 1984.

[6] S. Begej, "Tactical Sensor using Optical Technology," COINS Technical Report, University of Massachusetts, Department of Computer and Information Science, Amherst, Mass., in preparation.

[7] J. Brinkman and H. G. J. Kuypers, "Splitbrain monkeys: Cerebral Control of Ipsilateral and Contralateral Arm, Hand, and Finger Movements," *Science*, vol.

176, no. 4034, pp. 36-539, 1972.

[8] J. F. Engelberger, *Robotics in Practice*, NY: AMACOM, 1980.

[9] R. S. Fearing, "Exploration of the Dextrous Hand Control Problem," *General Electric Technical Report 82CRD337*, December 1982.

[10] J. J. Gibson, *The Senses Considered as Perceptual Systems*, Boston: Houghton-Mifflin, 1966.

[11] G. Gordon, *Active Touch*, Oxford: Pergamon Press, 1978.

[12] A. R. Hanson and E. M. Riseman, "VISIONS: A Computer System for Interpreting Scenes," in A. R. Hanson and E. M. Riseman (Eds.), *Computer Vision Systems*, NY: Academic Press, 1978, pp. 303-333.

[13] C. Hewitt, "Viewing Control Structures as Patterns of Passing Messages," *Artificial Intelligence*, vol. 8, pp. 323-364, 1977.

[14] J. M. Hollerbach, "An Oscillation Theory of Handwriting," *Biol. Cybern.*, vol. 39, pp. 139-156, 1981.

[15] D. H. Hubel and T. N. Wiesel, "Sequence Regularity and Geometry of Orientation Columns in the Monkey Striate Cortex," *J. Comp. Neurol.*, vol. 158, pp. 267-294.

[16] M. Jeannerod, "Intersegmental Coordination During Reaching at Natural Visual Objects," in J. Long and A. Baddeley (Eds.), *Attention and Performance, IX*, Hillsdale: Erlbaum, 1981, pp. 153-168.

[17] M. Jeannerod, "The Timing of Natural Prehension Movements", *Journal of Motor Behavior*, in press.

[18] M. Jeannerod and B. Biguer, "Visuomotor Mechanisms in Reaching within Extrapersonal Space," in D.J. Ingle, M.A. Goodale and R.J.W. Mansfield (Eds.), *Advances in the Analysis of Visual Behavior*, Cambridge: MIT Press, 1982, pp. 387-409.

[19] M. Jeannerod, B. Biguer and J. Michel, in press.

[20] E. R. Kandel, "Central Representation of Touch," in E.R. Kandel and J.H. Schwartz (Eds.), *Principles of Neural Science*, NY: Elsevier/North Holland, 1981, pp. 184-198.

[21] I. A. Kapandji, "Biomechanics of the Thumb," in R. Tubiana (Ed.), *The Hand*, vol. 1, Phila: W.B. Saunders and Co, 1981, pp. 404-422.

[22] J. M. F. Landsmeer, "Power grip and precision handling," *Ann Rheum Dis.*, vol. 21, pp. 164-170, 1962.

[23] D. Lee, "Visuo-motor coordination in space-time," in G. Stelmach and J. Requin (Eds.), *Tutorials in Motor Behavior*, Amsterdam: North Holland, 1980.

[24] L. I. Liebermann and M. A. Wesley, "AUTOPASS: An Automatic Programming System for Computer Controlled Mechanical Assembly," *IBM J. Res. Develop*, July 1977.

[25] C. Long, P. W. Conrad, E. A. Hall and S. L. Furler, "Intrinsic-Extrinsic Muscle Control of the Hand in Power Grip and Precision Handling," *The Journal of Bone and Joint Surgery*, vol. 52A, no. 5, pp. 853-867, 1970.

[26] T. Lozano-Perez, "Automatic Planning of Manipulator Transfer Movements", *IEEE Trans. Systems, Man, and Cybernetics*, vol. SMC-11, no. 10, October 1981.

[27] J. Y. S. Luh, "An Anatomy of Industrial Robots and Their Controls," *IEEE Trans. on Automatic Control*, vol. AC-28, no. 2, pp. 133-153, February 1983.

[28] R. Malek, "Prehension" in R. Tubiana (Ed.), *The Hand*, vol 1, Phila: W.B. Saunders and Co, 1981, pp. 477-480.

[29] D. I. McCloskey and S. C. Gandevia, "Role of Inputs from Skin, Joins and Muscles and of Corollary Discharges, in Human Discriminatory Tasks," in G. Gordon (Ed.), *Active Touch*, Oxford: Pergamon Press, 1978, pp. 177-187.

[30] J. Napier, "Prehensile movements of the human hand," *J Anatomy*, vol. 89, pp. 564, 1955.

[31] J. Napier, "The Prehensile Movements of the Human Hand," *Journal of Bone and Joint Surgery*, vol. 38B, no. 4, pp. 902-913, 1956.

[32] U. Neisser, *Cognition and Reality: Principles and Implications of Cognitive Psychology*, San Francisco: Freeman and Co, 1976.

[33] K. Overton, "The Acquisition, Processing, and Use of Tactile Sensor Data in Robot Control," Thesis submitted for the degree of Ph.D., Dept. of Computer and Information Science, University of Massachusetts at Amherst, 1984.

[34] R. P. Paul, *Robot Manipulators: Mathematics, Programming, and Control*, Cambridge, Mass: MIT Press, 1981.

[35] J. K. Salisbury, "Kinematic and Force Analysis of Articulated Hands", Thesis submitted for the degreee of Ph.D., Department of Computer Science, Stanford University, 1982.

[36] W. Schiff and E. Foulke, *Tactual Perception: A Sourcebook*, Cambridge: Cambridge University Press, 1982.

[37] G. M. Shepherd, *Neurobiology*, NY: Oxford University Press, 1983.

[38] R. Tubiana, "Architecture and Functions of the Hand," in R. Tubiana (Ed.), *The Hand*, vol. 1, Phila: W.B. Saunders and Co, 1981, pp. 19-93.

[39] A. J. Wright, "End Effector Technology and Programmed Automatic Exchange," in *13th International Symposum on Industrial Robots and Robots 7*, Chicago, Illinois, April 17-21, 1983, pp. 18.1-18.13.