Embodiment and Manipulation Learning Process for a Humanoid Hand

by

Yoky Matsuoka

B.S., University of California, Berkeley (1993)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1995

Signature of Author ....
Department of Electrical Engineering and Computer Science
May 12, 1995

Certified by ....
Rodney A. Brooks
Professor, Department of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by ....
Frederic R. Morgenthaler
Chairman, Departmental Committee on Graduate Students

Embodiment and Manipulation Learning Process for a Humanoid Hand

by

Yoky Matsuoka

Submitted to the Department of Electrical Engineering and Computer Science
on May 12, 1995, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

Babies are born with simple manipulation capabilities such as reflexes to perceived stimuli. Initial discoveries by babies are accidental until they become coordinated and curious enough to actively investigate their surroundings. This thesis explores the development of such primitive learning systems using an embodied light-weight hand with three fingers and a thumb. It is self-contained having four motors and 36 exteroceptor and proprioceptor sensors controlled by an on-palm microcontroller. Primitive manipulation is learned from sensory inputs using competitive learning, back-propagation algorithm and reinforcement learning strategies. This hand will be used for a humanoid being developed at the MIT Artificial Ingelligence Laboratory.

Thesis Supervisor: Rodney A. Brooks
Title: Professor, Department of Electrical Engineering and Computer Science

Acknowledgments

Shhhh, don't wake me up yet. I am still dreaming about writing a 90 page thesis. Before this whole thesis evaporate, I would like to thank many many people who made this dream possible, and let me print it out so that there will be a proof.

First, the bad boy of robotics [19], Rod Brooks. He was always there for me no matter what I said or did. He was an incredible advisor when I needed help with work, and a totally awesome friend when I needed him to be a buddy. Sometimes you feel lucky for meeting someone special in your life, and he made me feel lucky and happy. Thanks, Rod.

And of course, my pleasure dome-mates, Cindy Ferrell and Matt Marjanovic. Without wasting 6 weeks together painting our office, I could not have possibly written this thesis. This happy environment kept me going!

The other Cog members, Matt, Scaz and Robert, thank you for putting up with my obnoxiousness.

The people who I played tennis and drank beer with, thank you for keeping me sane!

All my friends, I love you all so much.....tears.....

And I would like to thank Okasama and Otosama (my mom and dad) the most for being the guidance in my life. And telling me not to work hard. Don't worry, I never do unless I have to!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Cognitive Challenge

As long as human beings have existed on earth, we have always attempted to decode or understand our brains. Our brain is more advanced than any other species which gives us capabilities to communicate and learn. The skill of communication also gives us the freedom to learn from others' experiences. Research in human cognition, formerly limited to the fields neuroscience, cognitive science, philosophy and psychology, has recently been extended to artificial intelligence where scientists attempt to recreate what is not known yet to our species.

In adults, almost 1 million motor neurons control our muscles[26], enabling an enormous range of complex activities. The primary motor cortex is known to be active when the body movements are detected. As shown in the somatotopic maps in Figure 1-1, disproportionally large sections of the motor cortex and the somatosensory cortex are responsible for representing the fingers and the hand. This results in our capability for intricate movements and precise sensing with our fingers.

However, babies are born with only reflexive capabilities for manipulative movements. A reflex is an involuntary, stereotyped response to a sensory input. For example, babies curl their fingers when the palm is stimulated. This capability, in

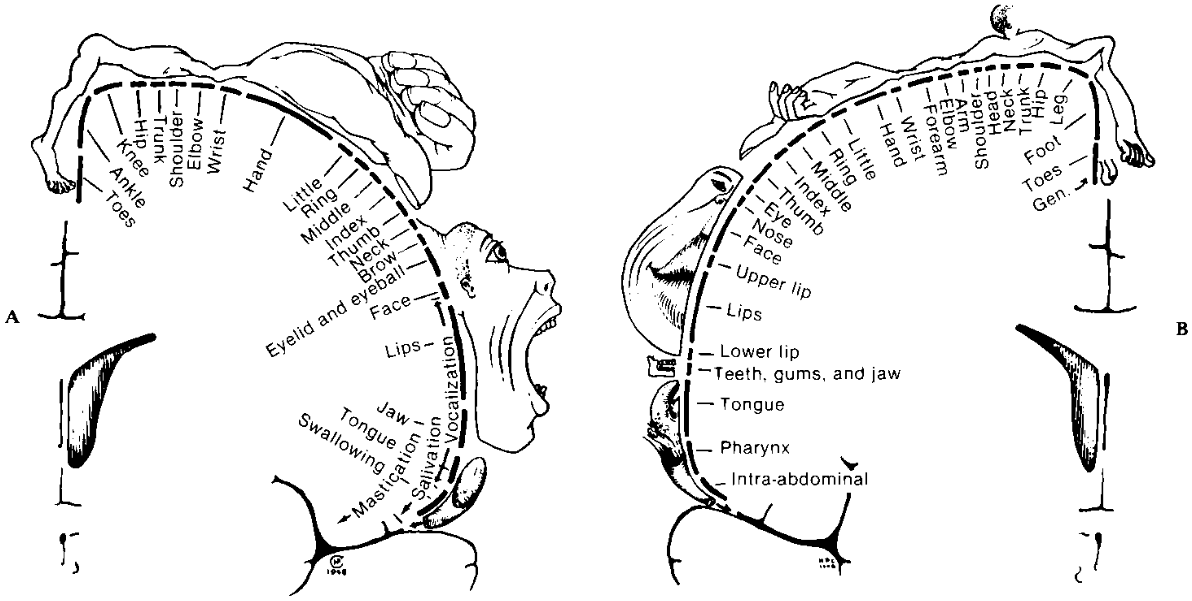

Figure 1-1: Human somatotopic mappings: left, motor cortex; right, somatosensory cortex.

conjunction with babies' curiosity and visual feedback is the bases from which they learn to manipulate objects and results in the eventual large portion of cortex mapping. Unfortunately it is still unclear why and how these connections develop in our brain. In neural and computer science, many learning strategies are developed based on our learning properties. However, they are full of assumptions and definitions that are not necessarily valid in the real world, such as Markov chain condition. As one of the steps, our approach to this complex phenomena is to reconstruct our behavior and study the learning process using our faster than ever calculation power of computers,in order to provide insight into the human's brain functionalities.

1.2 The Physical Challenge

Sensory information is first detected by the receptors which is routed and processed within the nervous system to interacts with the brain. Mechanoreceptors, receptors that respond to physical deformation, are responsible for touch and pain. The ridges on the fingers orient cutaneous mechanoreceptors called Meissner corpuscles, and they are largely responsible for our ability to perform fine tactile discriminations with our fingertips. The receptors monitor the environment and transduce the information which is then propagated and passed toward the spinal cord. The spinal cord, being only about 42 cm long $\times$ 1 cm diameter, receives all the motor and sensory inputs, which are fed into multiple ascending sensory pathways and local reflex circuits. Unfortunately, the current connector and wire technology does not allow us to build such a system due to the size and inorganic material limitations. Even when only one hundred 30 gauge stranded wires are run through a small joint and repetitive strain is applied, the wires are prone to breakage due to the inflexibility characteristics of conductive materials.

The human body is adaptable to situations and tasks which can be learned through training using the same physical body parts. To date, most mechanical hands and grippers constructed are task driven and limited to performing a very few specific tasks. They may excel in their precision and strength for a particular task, but their inflexibility to perform non-specified tasks make the existing hands nonhuman. The human hand is an amazing device, capable of manipulating diverse objects and tasks, yet its precision and srength requires more external muscular assistance, feedback, and training than we imagine. The challenge is to build a system that is not preconfigured, but is able to learn to accomplish many tasks like our hands.

1.3 Terminology

Many parts of hands and fingers and discussed through out this thesis. For a simplicity, the terminolgies are based on the human anatomy terminologies shown in Figure 1-2[24].

The mechanical hand constructed for this thesis have three fingers, each having two segments and two joints, and a thumb with one segment and a joint, so the terms

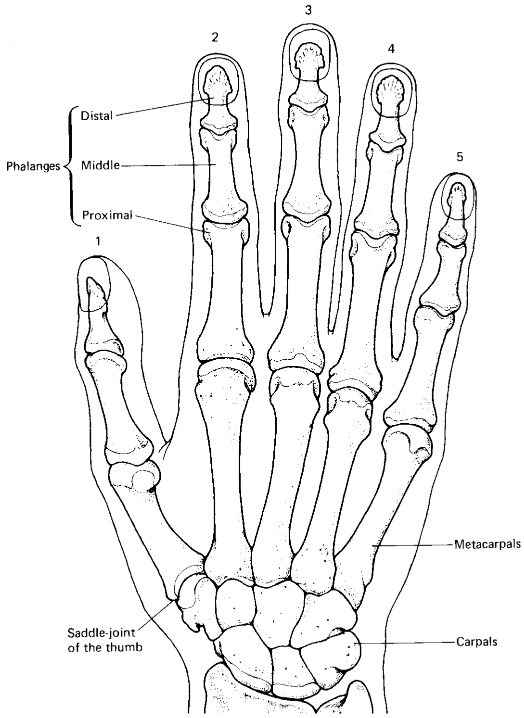

Figure 1-2: Human anatomy terminology.

are altered as shown in Table 1.1.

1.4 Organization of Thesis

This thesis is organized into 4 additional chapters as follows:

Chapter 2 discusses the motivation for embodiment in this project. It introduces the behavior of humans and related research previously done which leads to the current humanoid research. It also argues the importance and difficulties of embodiment. Embodiment is one of the best approaches in order to learn about human cognition, but due to mechanical dufficulties, many constraints are considered.

Chapter 3 presents a detailed description of the hand built for this research. It covers the mechanical design and implementation including the structure of physical hand, tendon cabling strategy, actuators, sensors and computing tools.

Area	*Part*	*Terminology*
fingers	2nd digit	index finger
	3rd digit	middle finger
	4th digit	ring finger
	segment further away from palm	Distal
	segment closer to palm	Proximal
	joint further away from palm	distal joint
	joint closer to palm	proximal joint
thumb	segment	Proximal
	joint	proximal joint
palm	inside	palm
	outside	dorsum
all	segments	phalanges
	joints	joints

Table 1.1: Terminologies of mechanical hand parts used.

Chapter 4 has two parts. The first part describes the PID controller which is used locally to incorporate the primitive motion of the hand. The second part presents the learning strategies which is inspired by an infant's learning process. Strategies such as competitive learning, back-propagation algorithm and reinforcement learning are introduced and implemented. The experimental results are also shown in this chapter.

Chapter 5 reviews the research discussed in this thesis and concludes with a discussion of the future work.

Chapter 2

Embodiment

This chapter presents the motivation for embodiment and illustrates its significance to this thesis. Humans' cognitive and physical behavior is discussed, with focus on infants' manipulation behavior. The hand built is a self-contained human scaled non-task driven tool learning its own cognitive and physical behavior, differentiating this research from previous work in manipulation tools. The advantages and disadvantages associated with building such a system are considered.

2.1 Motivation and Related Work

2.1.1 Infants

Piaget was one of the first of the modern psychologists to recognize the infant's manipulative exploratory behavior with the environment as a vehicle of cognitive stimulation[22]. Infancy is not only a time when muscles and the nervous system mature, but also a time of active and continuous learning which allows a baby to establish effective transactions with the environment and move toward a greater degree of autonomy. During this time, infants practice and perfect sensorimotor patterns that become behavioral modules which will be seriated and imbedded in more complex actions.

Human motor control is a sequential process which is affected by the order of development of different regions of the brain and the nervous system. Since the control of the central body areas matures before the outer areas, hand development comes later than for other parts of the body. Consequently arm motion controlled by a more mature shoulder joint causes accidental collisions with objects in the environment as the infants come in contact with an increasing number and variety of objects. Reflex is the only hand motor control present at birth. When the skin of the palm is touched by an object, the muscles of the hand contract and results in curling the fingers, whereas if a strong force is applied, the fingers open to alleviate pain by expanding the muscles. The reflex is completely controlled and pre-programmed at the spinal cord and the summary of the reaction reaches the brain long after the action has been taken. When this process repeats itself, the nervous system makes the connection between the stimulus and its corresponding actions, resulting in the first step of manipulation learning. Through touching the objects, babies learn their shapes, dimensions, slopes, edges, and textures. They also finger, grasp, push, and pull to learn the material variables of heaviness, mass and rigidity, as well as the changes in visual and auditory stimuli that objects provide. Visual feedback is a crucial piece in manipulation learning as seen in the infants of a few days old extending their hands toward a visible object[28]. This instinctive motivating information is triggered somewhere in the nervous system and allows explorative learning to initiate.

2.1.2 Mechanical Hand

Since the eighteenth century the mechanics of hands has been studied and has been the model for various mechanical constructions, primarily for protheses and telemanipulators, manipulators controlled remotely[21]. More recently, human hands have been analyzed for industrial mechanical grippers and many of them are used reliably in assembly settings. They are built specifically for the environment in which the grippers have to work, and they are so different for each application that a standard industrial hand that satisfies every need cannot be built. Their functions are mostly clamping, vacuum, and magnet which are activated by pneumatic, hydraulic, electric and mechanical force.

The first dexterous mechanical hand that resembled a human hand was the Utah/MIT Dextrous Hand built about 10 years ago[14]. The hand itself was approximately anthropomorphic in size, including three tendon operated fingers and a thumb with multichannel touch sensing capability. Each finger included three parallel axis joints and a proximal joint which are independently controlled using a tendon system totalling eight tendons and actuators per finger. 38 actuators are mounted in the forearm for controlling the tendon, and a pneumatic approach is used due to its low weight and compactness. Optical fibers and birefringent materials were used for their touch sensing system. The control system simply delivered joint angle commands to servo systems at each joint so that the hand assumed various desired configurations integrating touch sensors and tendons. This work was significant in a way that it could be used for multiple purposes in research, giving the capability to inegrate additional systems such as learning algorithms or more sensors in an anthropomorphic way.

2.1.3 Humanoid

Attempts

Originally, most humanoid robots were clever adaptations of existing industrial robots or specialized mechanical arms. Later there were explicit attempts to make robots anthropomorphical in appearance and capabilities. Wabot was exhibited at the Japanese Expo in 1985 and it played a piano, with its precise and fast finger works[32]. It had a human appearance and if examined briefly, it could visually fool people that it had a cognitive system. Though this robot design was inspired by the human hand motor system, it was not practical in any sense of the word. It was bolted in front of a piano, and the only capablility it had is to play a piano. No other tasks,

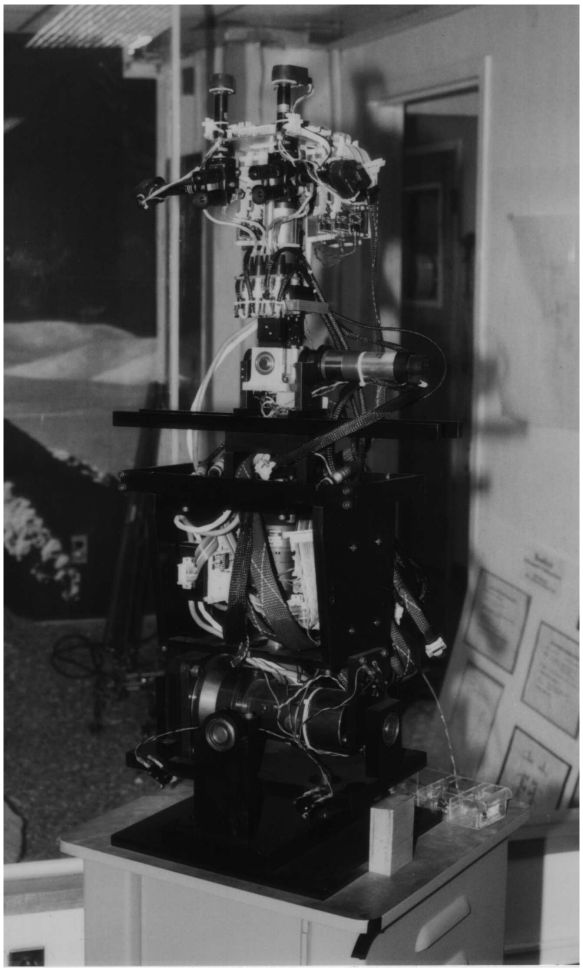

Figure 2-1: A picture of Cog.

even just to manipulate objects, could have been done by the robot. While various engineering enterprises have modeled their artifacts after humans to one degree or another, nobody seriously tried to couple human like cognitive processes to these systems methodologically.

Cog

At the MIT Artificial Intelligence Laboratory, a research group headed by professors Rodney A. Brooks and Lynn Andrea Stein is currently developing an integrated physical humanoid robot named Cog [3] shown in Figure 2-1. This system will include vision, sound input and output, and dextrous manipulation all controlled by a continuously operating parallel MIMD computer as the brain. The processors are 16Mhz Motorola 68332s in standard boards which plug 16 to a backplane. The backplane provides each processor with six communications ports and a peripheral processor port. It has the capability to connect up to 256 processors by stacking 16 backplanes to a single front end processor. Each 68332 communicates up to 16 Motorola 6811s which are single chip processor with onboard memory, timer, SPI, analog to digital convertor, and some I/O ports. The motor skills that are handled at the spinal level for humans are processed by 6811 motor boards to act like spinal cords. The goals of this project are to build a prototype general purpose autonomous robot and to understand human cognition. This is the first time anyone has attempted to construct an embodied autonomous humanoid intelligent robot.

Currently we are at a primitive building and integrating stage in hardware and software including arms, hands, ears and eyes. As we put the pieces together we will be forced to understand the physical constraints which can lead to a better understanding of how we should build the pieces. When all the parts are integrated to our one front end processor, we will be able to treat Cog as a whole to attack problems that require coordinating the whole body. A simple operation, such as picking up a bell, requires sound localization, torso control, visual feedback and arm and hand manipulation skills. This kind of task may only be done at the cognitive level using a system like what we are building right now. Cognitively, this project is important because studying the way Cog decides to execute certain actions may lead to an understanding of our own cognition.

2.2 Embodiment of Hand

2.2.1 Overview

Why?

The importance of embodiment in order to understand human cognition is a controversy in the artificial intelligence community. Many argue that a simulation of such a system can satisfy the need, and would not waste the time needed to build a complex hardware creature. We live in a noisy environment and we are capable of learning to ignore irrelevant noise. For example, we can recognize a telephone even when the edges are dirty or chipped, which cannot be easily done with the current computer vision technology. We are restricted by the limited technology that allows us to build such a system, but also limited by what we know about human biological systems.

Another example to show the importance of embodiment is the study of bird wings. The physics of bird wings have been studied to embody in a human scale with our dream to fly since the 16th century. With a solid understanding of aerodynamics, a computer simulation can be built to understand the fuctionality of the wings better. Even for a simulating such a simple environment as air, many assumptions such as wind and pressures need to be made in order for the simulation to work consistently. While studying such a system can show important points in the flying mechanism, the system still need to be physically built to understand other constraints that occur only in the real world setting.

The attempt to understand human functionality is much more complicated than studying bird wings. Many assumptions, probably including some that are not valid in the real environment, are necessary because we do not know enough about how we process information that we receive from the environment. Therefore, it is more crucial to build such a system physically to understand its constraints and limitations. As we build the system, still with many assumptions and using existing technology, we may realize human's functionalities that simulations have not been able to teach

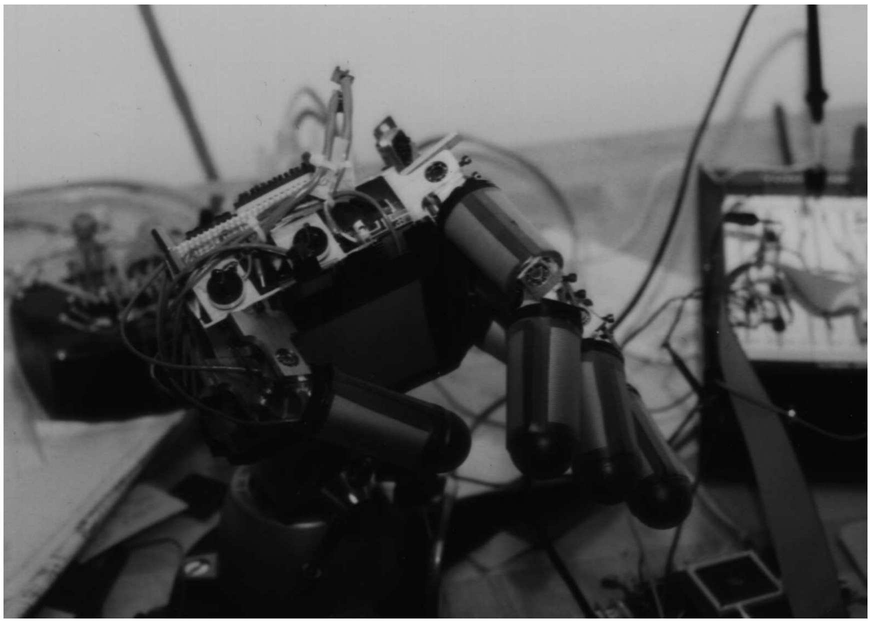

Figure 2-2: A picture of Cog's hand.

us. By attempting to build an anthropomorphic hand, many physical limitations and constraints are realized, and those realizations are requisite to unraveling the questions of human physical and cognitive functions.

Physical Setting

This project uses an anthropomorphic scale hand which has three fingers and an opposing thumb shown in Figure 2-2. Each finger has two coupled joints that are controlled by a miniature steel cable. Due to the nature of a coupled cabling strategy, it is compliant. There are four motors controlling each finger, generating a maximum torque equivalent to holding a 0.5 pound object at the tip of a finger. Motors are integrated with rotational potentiometers to detect the motor positions. Force/pressure sensors cover the surface of all fingers. A finger has two phalanges and each of it has two force sensors. The thumb has two force sensors and the palm has four position sensors in addition to a force sensor. All the sensory readings are multiplexed and converted to digital signals at a Motorola 6811 microcontroller which is integrated on the top of the dorsum. The 6811 has four analog-digital converter ports, four pulse width modulator ports which are connected to the motor drivers and all four motors. The microcontroller acts like a spinal cord, containing a PID control loop and handling reflexive reactions. A larger microcontroller 68332 is interfaced for higher operations such as learning and coordinating with other features such as eyes and ears.

2.2.2 Constraints

Strength and Precision

Many researchers have successfully created hands that are reasonably small and strong, interfaced with large forearms in order to carry many high-powered motors, precision encoders, and gears [2, 30, 31]. However, in creating a human scale model, it is crucial to minimize the weight and the size of the hand. As a trade off, increasing the strength and the precision becomes complex. Minimizing wires and cables is achieved by placing actuators close to the joints, and local processors close to all the sensors and motors. Optimally everything should be contained within the fingers and the palm. In order to contain motors in the hand, both the number and the size of motors need to be decreased significantly from all the existing mechanical hands. To reduce the number, all the joints in a finger are coupled with a tendon cable which is pulled from both directions for curling and expanding by a single motor. This strategy limits the strength of the hand due to the conciseness of the motors, the compliancy of the cabling strategy, and the material of cables. To avoid using large encoders, rotational potentiometers are used at the expense of reducing the accuracy from 16 bits to 8 bits of information. Needless to say, small parts are difficult to construct, which increases the complexity of these constraints. Though, when the human hand mechanism is analyzed for infants, each finger has minimal torque and it is impossible to even estimate the angles of the joints without visual feedback or external applied force. Thus, studying infants' learning skills only requires the strength of industrially available miniature motors and the precision of potentiometers read through 8 bit analog to digital converters.

Stability and Orientation

For multi-finger manipulators, stablizing the grasp is a critical issue. According to some investigations done in the past, a four finger manipulator can handle 99% of the parts that a five finger manipulator or human hand can handle, a three fingers can handle 90% and two fingers 40%. For the humanoid hand, a three finger with a thumb configuration is used to reinforce the stability for various shaped object manipulation [30]. For example, the last finger can be used as the base to hold a small object. Young infants do not use the thumb as an opposing finger, and use all fingers like a one degree of freedom compliant gripper. As learning proceeds, the opposing thumb becomes the most important finger for manipulation and slowly increases the degrees of freedom to more than twenty-five, though many are coupled by the nature of the ligament structure and location of tendon insertions. For our embodied hand, all four fingers have a designated motor which gives each one degree of freedom. However three fingers have two coupled joints yielding a total of seven degrees of freedom visually. From the construction of the hand, various objects can be manipulated within the torque limit of the arm and the hand.

The orientation of the hand during reaching is an important part of a grasping procedure. Babies' initial reaches are awkward, but learn to coordinate and turn it into a smooth movement within a few months[34]. The initial reaction during reaching is to orient the palm toward the desired point of contact, and preshape the fingers according to the shape of the object. Unfortunately, without visual feedback or arm movement coordinating with the hand, those precedures need to be ignored. For this research, the orientation of the hand is fixed to have the palm perpendicular to the ground for simplicity.

2.2.3 Sensorimotor System

Meissner corpuscles are elongated encapsulated endings that are oriented with their long axis perpendicular to the suface of the skin. They are quite numerous in the skin of fingertips, and they are largely responsible for our ability to perform fine tactile discriminations with our finger tips. Unfortunately, this system is still not well enough understood to implement it to an inorganic form. Tactile sensing research is an ongoing field where any commercially available skin is not good enough yet to be interfaced to achieve human like precision. For creating a human like system, many constraints need to be considered to find an optimal solution within our existing technology. First, the skin needs to be flexible to adopt the shape of the surface of fingers and palm. Second, the size and the number of wires needs to be minimized for creating a human scaled hand. The phalanges are hollowed to allow wires to run through them, but it is still a very limited space.

One of the main goals of this project is to learn from building a cognitive system and learn how such a system should be built. For this purpose, we can start off using a tactile system that is not as accurate as human finger skin. If the cognitive system we build tells us in the future that more precise tactile sensors play a crucial role for learning, we will try to add such a system. Since the most important information needed is the force information followed by the position of contact, many force sensors and several positions sensors are used for the hand.

2.2.4 Learning

Learning manipulation in an unpredictable, changing environment is a complex task. It requires a nonlinear controller to respond in a nonlinear system that contains significant amount of sensory inputs and noise[23]. Investigating the human manipulation learning system and implementing it in a physical system has not been done due to its complexity and too many unknown parameters. Conventional adaptive control theory assumes too many parameters that are constantly changing in a real environment [33, 37]. For an embodied hand, even the simplest form of learning process requires more intelligent control network. Wiener [36] has proposed the idea of "Connectionism", which suggests that a muscle is controlled by affecting the gain of the "efferent-nerve – muscle – kinesthetic-end-body – afferent nerve – central-spinal-synapse – efferent-nerve" loop. Each system within the loop such as efferent nerve contains its own feedback loop system. This kind of loop is inherently nonlinear with the capability to take many noisy inputs and may be implemented in a physical hand. It is still very limited to what kind of learning strategies can be used for an implementation, but as an individual system, standard competitive learning and backpropagation algorithms are used. To connect the whole system, a connectionist implementation of reinforcement learning is used for the embodied hand.

Chapter 3

Hardware Design

This chapter presents the hardware design of Cog's hand. The hand is made of aluminum and designed to minimize weight and size. It has a microprocessor and sensor interface circuit on top of the dorsum and has 36 total sensors on the surface and joints.

3.1 Mechanical Design

3.1.1 The Structure of Hand

The hand has a 4.0 inch $\times$ 4.0 inch palm with three fingers and an opposing thumb where the diameter of fingers is 1.0 inch. To minimize the weight and allow for space to run cables and wires, each phalange is hollowed out using a lathe to 0.02 inch thickness. Joint design is done as in Figure 3-1 by setting,

$$\max(\theta) = 95^\circ \tag{3.1}$$

$$\max(\phi) = 90^\circ \tag{3.2}$$

where θ is the angle for the proximal joint and ϕ for the distal joint. There are physical limits at the proximal joint and the distal joint as shown in Figure 3-2 so

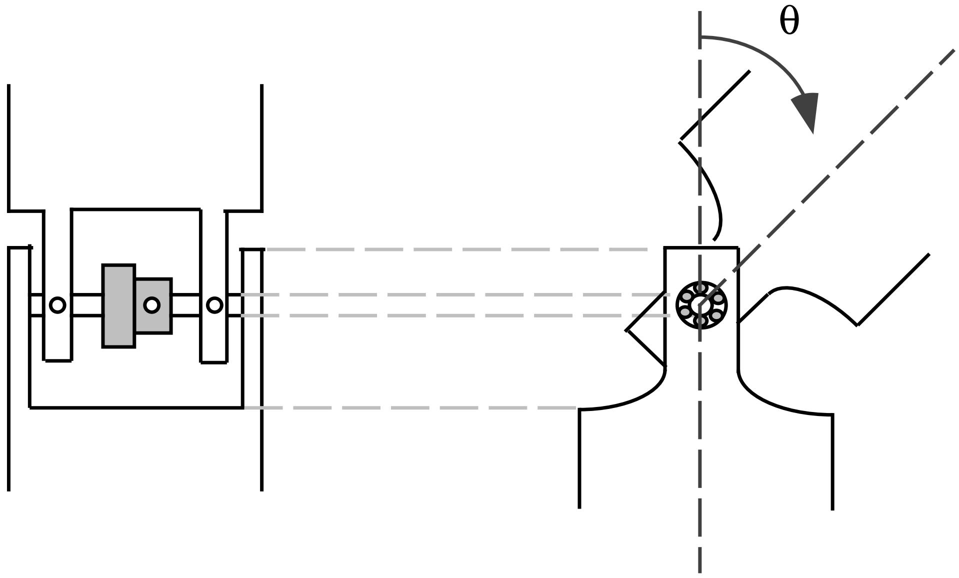

Figure 3-1: A diagram of joint with a bending angle: frontal view(left), side view(right).

that when fingers are fully open,

$$\min(\theta) = -5^\circ \tag{3.3}$$

$$\min(\phi) = -10^\circ. \tag{3.4}$$

Within a joint, there is a miniature steel pulley of diameter 0.5 inches and a shaft that is fixed to the phalange above the joint(i.e.,the pulley in distal joint is fixed to Distal), and friction is minimized using miniature ball bearings. Cables are run in such a way that both curling and expanding are controlled using one continuous cable and one motor as shown in Figure 3-3. This cabling mechanism works because the rotational force applied by a motor results in a tension in the cable that causes the friction force of the pulleys to move the joints(Figure 3-4). The steps of applied and induced force effects of this mechanism are illustrated using a finger curling example:

1. Motor applies a tension to the inner cable.

2. Friction2 becomes strong enough to rotate the pulley in the proximal joint.

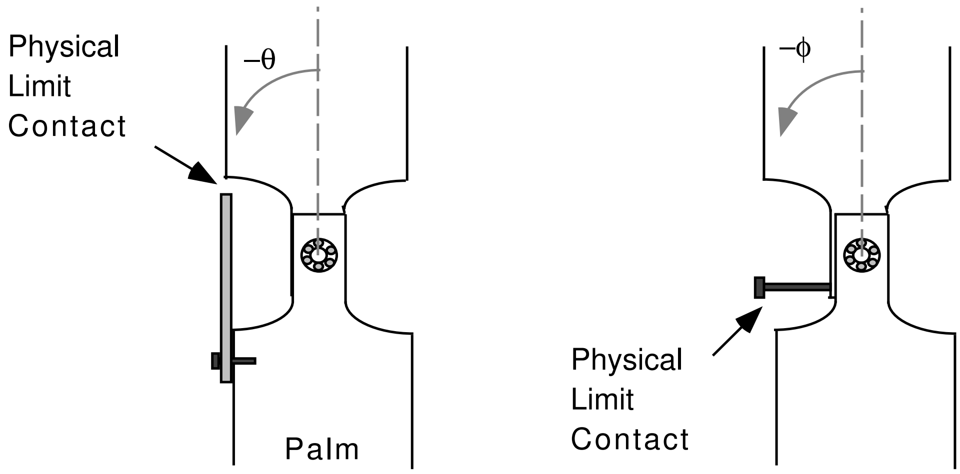

Figure 3-2: Physical limits for a proximal joint(left) and a distal joint(right).

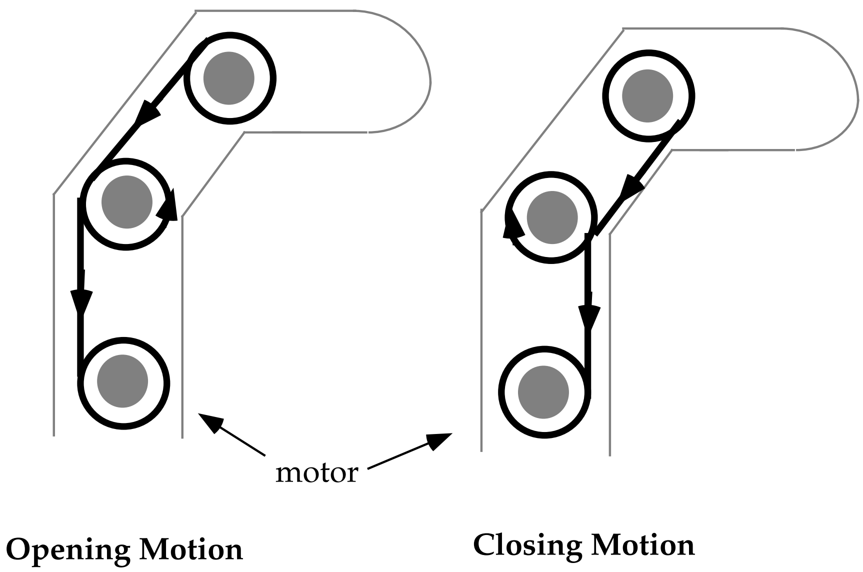

Figure 3-3: Cabling configurations of curling and expanding motion.

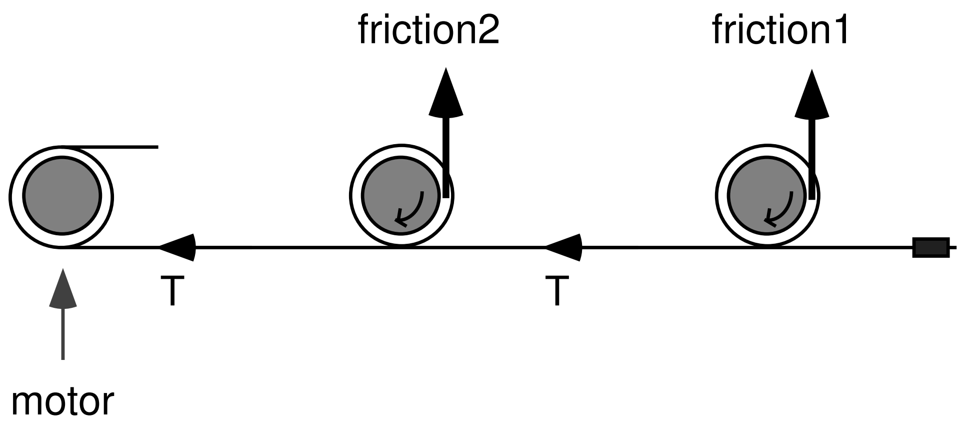

Figure 3-4: Theory of cabling mechanism with applied and induced forces.

3. Proximal comes in a contact with an object or reaches a physical limit causing a resistive force.

4. The resistive force from step 3 overcomes friction2 causing the cable to slip over the pulley in the proximal joint applying tension in the cable in Proximal.

5. Friction1 is induced to rotate the pulley in the distal joint.

6. Distal comes in a contact with an object or reaches a physical limit causing a resistive force.

7. The cable reaches its maximum tension and stops. This is an optimal grasping configuration for this finger.

To achieve such a coupling effect for the joints, the tension of the cable and the potential friction for the surface of pulleys need to be considered in detail. If the pulley potential friction is too high, the resistive force at step 3 cannot overcome the friction and the distal joint could not be controlled. When the cable tension is higher than required as the finger curls, the proximal joint is controllable whereas the distal joint cannot be moved. The proximal joint is still controllable because the tension of the inner cable applied by the motor is larger compared to the force against it due to a minor slip of outer cable that occurs within the proximal joint to alleviate the

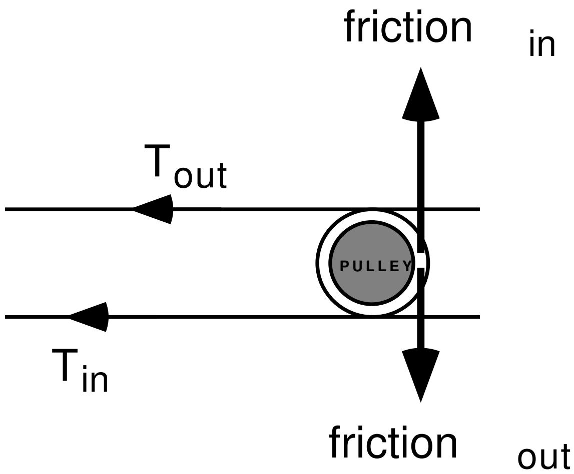

Figure 3-5: A free-body diagram for a distal joint pulley.

tension between the motor and the proximal joint. Due to this effect, the tension of the outer cable between the distal joint and the proximal joint increases and induces the frictional force against the direction of friction that causes the joint movement as shown in Figure 3-5. As a result, there is not enough friction to move the distal joint. When the tension is too low, the compliance becomes too large to weaken the grasping force. The optimal total cable length is calculated using the formula,

$$\frac{L}{1.04} = 2(l_{Distal} + l_{Proximal} + z) + \frac{11}{2}\pi d \tag{3.5}$$

where L is the total cable length, l_m is the length of m, z is the length between the distal joint to the cable terminal point and d is the diameter of pulleys. $0.04L$ is added to achieve an optimal tension and compliancy. The material of the cable, nonstretching nylon coated steel is chosen for its durable characteristics, but it still stretches over time. A tension cranker is designed as shown in Figure 3-6 so that tension can be adjusted to an optimal strength when the cable is stretched over time. The cable is terminated using cable locks within Distal as shown in Figure 3-7.

At the palm, the fingers are separated by 0.5 inches and the outer fingers are fixed to the palm at 15 degrees away from the middle finger. Each finger has two phalanges

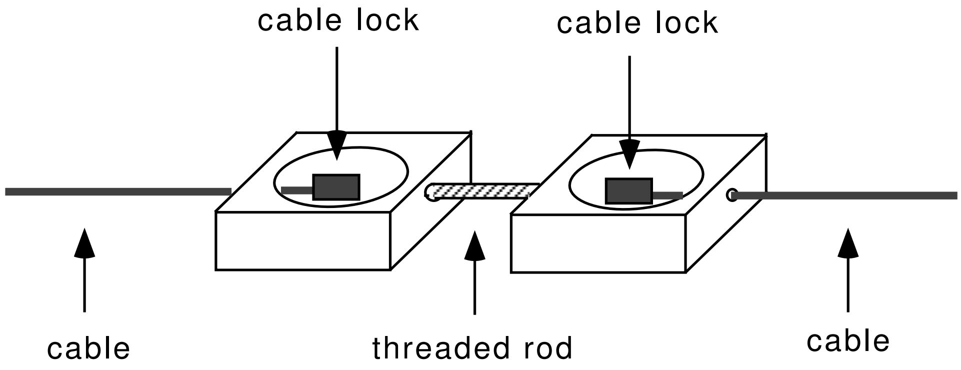

Figure 3-6: Tension cranker design for adjusting the stretched cable length.

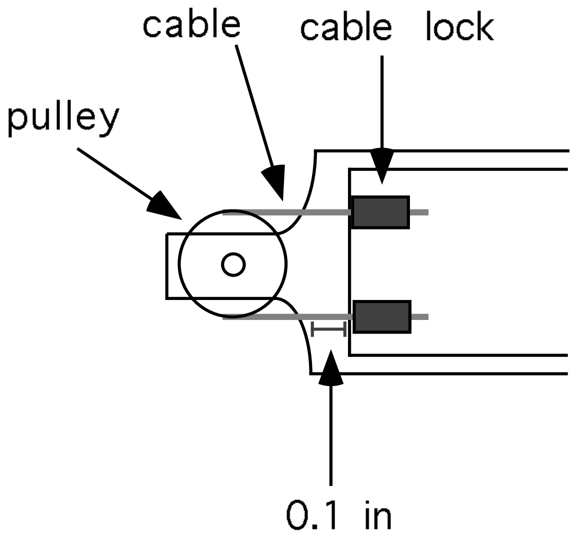

Figure 3-7: Cable termination using cable locks.

and their lengths are chosen to avoid colliding with other fingers, yet allowing the tips of all fingers to meet at one point when they are fully closed, which is shown in Figure 3-8. Using this figure, Distal and Proximal lengths are calculated using,

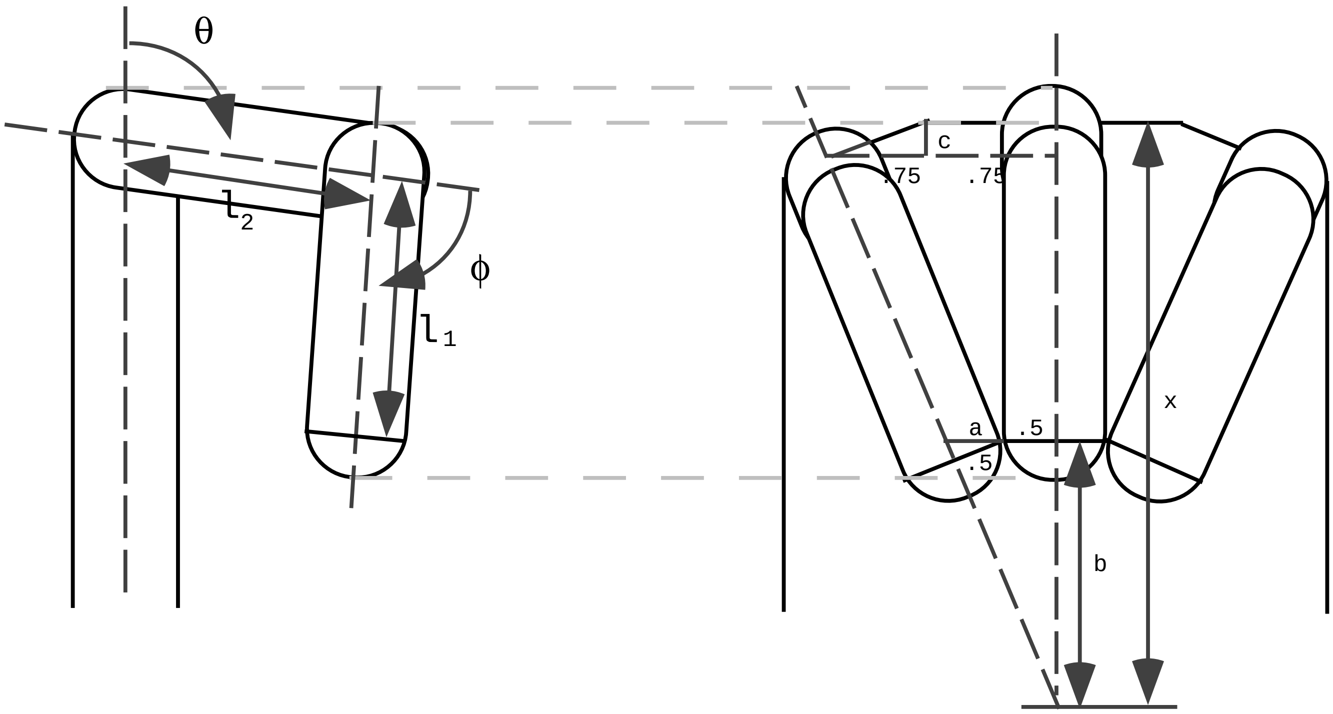

Figure 3-8: Diagram of hand used to determine the length of phalanges: sideview(left) and frontview(right).

$$x = l_1|\cos\theta| + l_2|\cos(\phi+\theta)| + b \tag{3.6}$$

$$b = \frac{a+0.5}{\tan 15^\circ} \tag{3.7}$$

$$a = \frac{0.5}{\sin 75^\circ} \tag{3.8}$$

$$x = \frac{0.75+0.75}{\tan 15^\circ} + c \tag{3.9}$$

$$c = 0.75\sin 15^\circ. \tag{3.10}$$

and having a total finger length to be 4.5 inches, a set of linear equations can be formulated to

$$l_1 + l_2 = 4.50 \tag{3.11}$$

$$0.087l_1 + 0.996l_2 = 1.999 \tag{3.12}$$

which gives the length of Distal to be 2.75 and the length of Proximal to be 1.75 and the phalanges are built accordingly. The tip of a finger is made of polyethylene, and covered with vinyl. An opposing thumb has one degree of freedom and the length is chosen to meet with other finger tips for the purpose of fine manipulation. It is fixed to the palm and the proximal joint is controlled with a steel cable as in the other joints. Because this joint is not coupled, the torque exerted for the thumb is larger than for the other fingers.

3.1.2 Motor Selection

There are four motors controlling each finger and they are contained within the palm (see Figure 3-9) to minimize the size. The motors and gearboxes were chosen by calculating the required torque and speed. At no load, the desired maximum angular velocity of the joints is 2 rps = 120 rpm, permitting the finger to open and close fully in 0.5 seconds. Considering finger's own weight and applied force, it is assumed that the overestimated maximum load is 1/2 pound centered one inch away from the motor. With this assumption, the stall torque is

$$\tau = 0.5\text{lbs} \times 1\text{in} = 16.0\text{oz-in.} = 0.12\text{Nm}. \tag{3.13}$$

Therefore the required power of the motor assuming 60 percent efficiency is

$$P = \tau\omega = \frac{0.12(4\pi)}{0.60} = 1.5Watts. \tag{3.14}$$

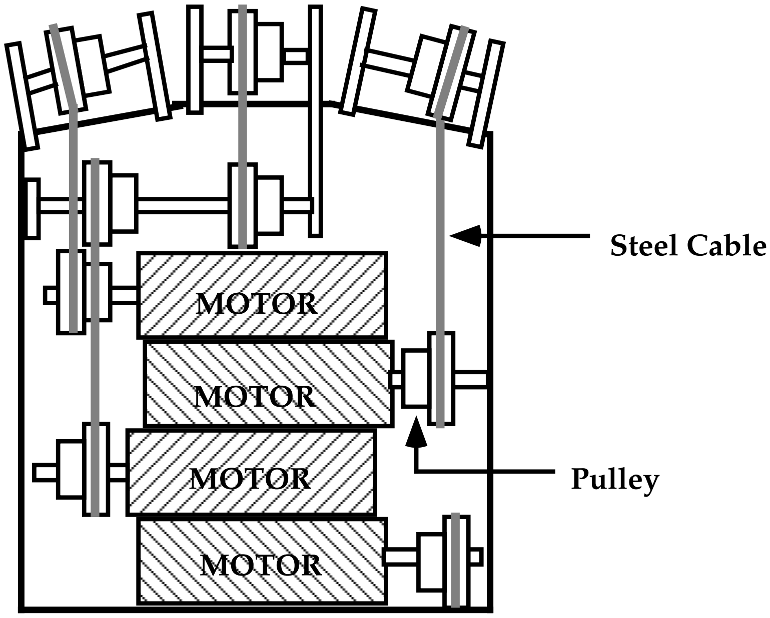

Figure 3-9: Inside the palm.

Maximum intermittent power output(Watts)	2.7
Maximum continuous power output(Watts)	2.0
Maximum efficiency(%)	76
No load speed(RPM)	11,300
Stall torque(oz-in.)	1.25
Maximum continuous torque(oz-in.)	0.35
Weight(oz)	0.71

Table 3.1: The characteristics of MicroMo's DC MicroMotor 1331.

To meet these criteria and to minimize weight and size, MocroMo's DC motor series 1331 with a 15/5 76:1 gearbox was chosen. The characteristics of the motor and the gearbox are shown in Table 3.1 and Table 3.2.

3.1.3 Grasping Capability

The size of objects to be manipulated is largely determined by the length of the phalanges. By taking advantage of the four fingered hand, large or non-trivial shaped objects may be grasped. For example, ring finger can be used as a base to hold a large object that other fingers cannot wrap all the way around. The manipulability is also

Reduction ratio	76 : 1
Maximum continuous output torque(oz-in.)	14.2
Maximum intermittent output torque(oz-in.)	42.4
Efficiency(%)	68
Weight(oz)	0.61

Table 3.2: The characteristics of MicroMo's gearhead 15/5.

dependent on the material of the object grasped. The surface of the hand is covered with a thin layer of vinyl to increase friction. When the static friction between the skin and the object overcomes the gravitational force, the object does not slip off. To analyze the friction, one point of contact with an object is considered. The static friction between the object and the skin is given by,

$$f \leq \mu_s N \tag{3.15}$$

where f is the frictional force, μ_s is the coefficient of static friction, and N is the magnitude of the normal force. Figure 3-10 shows the object at the moment that sliding is about to take place. The forces that act on the object are the normal force, N, that is the grasping force applied by fingers pushing into the object, the weight of the object W, and the frictional force, f. Because the object is in equilibrium, the resultant external force acting on it must be zero,

$$\sum F = f + W + N = 0. \tag{3.16}$$

The x component of this vector equation gives,

$$\sum F_x = f - W = 0. \tag{3.17}$$

At equilibrium, the static frictional force has its maximum value. Using Equation 3.15 and Equation 3.16, we get

$$f = \mu_s N = W. \tag{3.18}$$

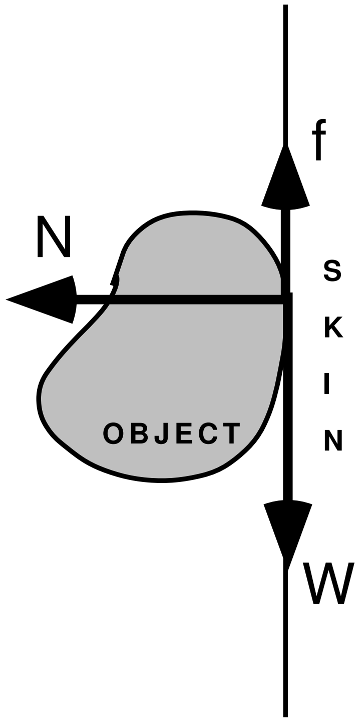

Figure 3-10: A free-body diagram of object on skin.

When an object is grasped, the finger is positioned using PID control such that a firm grasp is achieved by having a constant N. μ_s is a combination of μ_{sl}, μ_s of latex and μ_{so}, μ_s of the object, and W is object dependent, therefore, a relationship,

$$\frac{\mu_{sl} + \mu_{so}}{W} = \text{constant} \tag{3.19}$$

can be achieved. When a learning tool is available such as described in Chapter 4, various grasping positions can be considered to improve the skill as infants do during their manipulation exploratory stage.

3.2 Computation Tools

3.2.1 Spinal Cord Level Computation

A motor board with a 6811 and a sensor interface board are mounted on a dorsum as shown in Figure 3-11. They function like a spinal cord by controlling finger movements such as reflexes. The Motorola MC68HC711K4 includes CPU, 24 Kbytes of

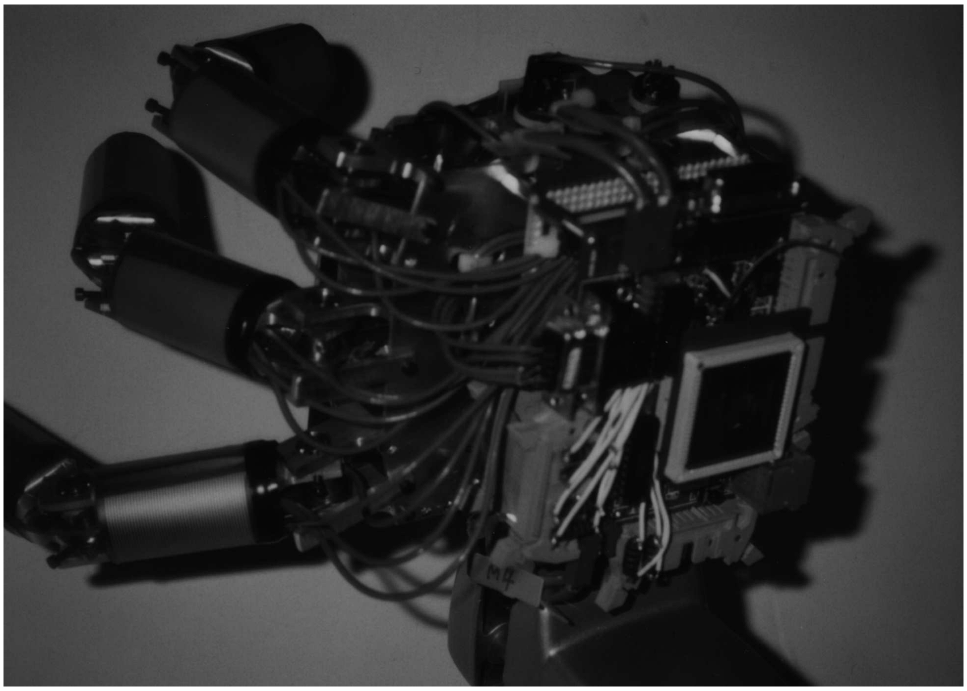

Figure 3-11: A picture of the dorsum with a motor board and sensor interface board mounted.

EPROM, 640 bytes of EEPROM, 768 bytes of RAM, four 8-bit pulse-width modulators, 8 channel 8-bit analog-to-digital converters, and other MC6811 features. The K4 has been chosen specifically to take advantage of onboard PWM pulsors with frequency and duty-cycle variations which allows the whole hand to be controlled by only one MC6811 chip and eliminate a complex sequence of latches and flip-flops. PWM frequency can be specified using two bytes between 0.05Hz to 40KHz using an 8MHz external crystal clock. The overall picture of the motor board design is

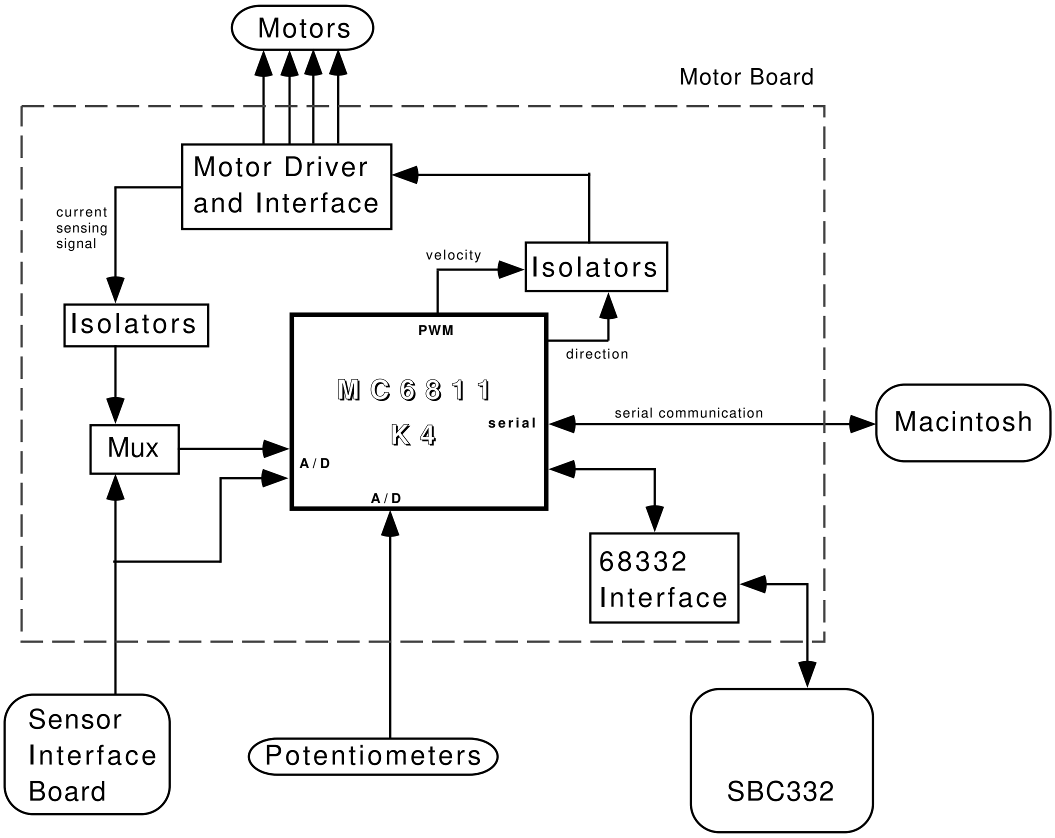

Figure 3-12: An overall picture of the motor board design using Motorola MC6811.

shown in Figure 3-12. The board is designed in a way that optoisolaters are used to isolate motor signals from analog/digital signals. A motor driver L293E takes a duty-cycled PWM signal and a direction, and sends a processed signal to a motor. The chip is also capable of sensing the load current which becomes part of the sensory information. The potentiometer outputs and the rest of the sensory information are multiplexed and fed to the analog-to-digital converter ports. The serial line is used to communicate with a CPU and download programs to EPROM and EEPROM, and the 68332 interface module decodes and connects to a SBC332 board that handles the brain level computation.

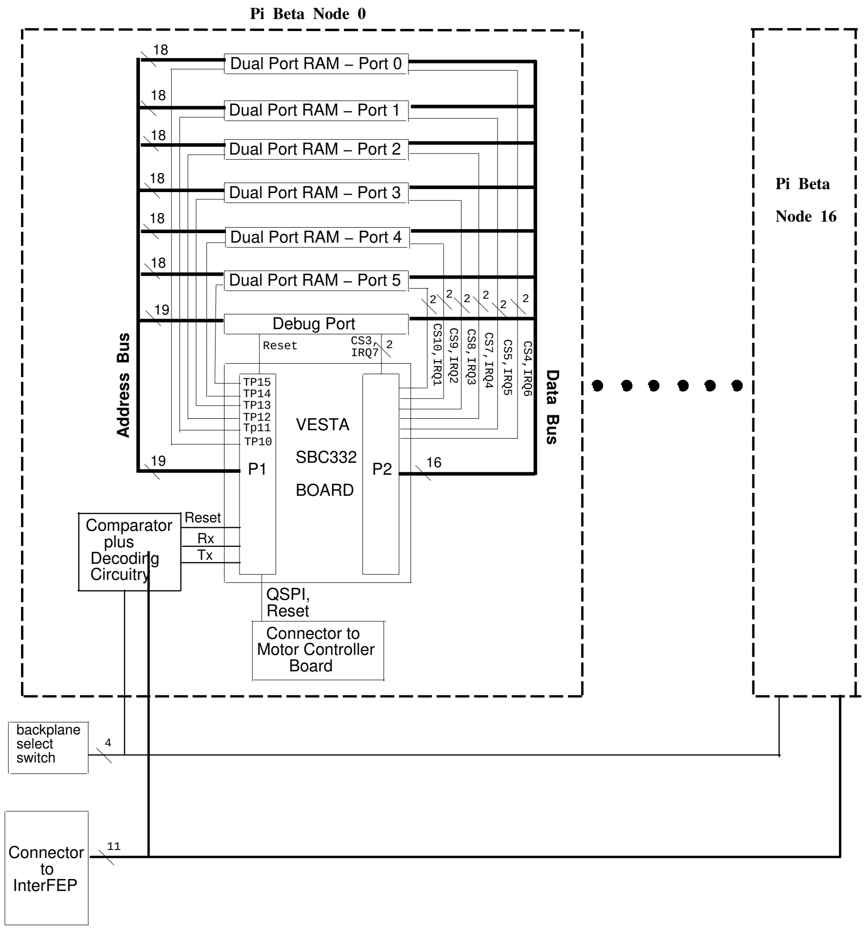

Figure 3-13: A backplane interfacing 16 processors.

3.2.2 Brain Level Computation

The computation done at this level is the massively parallel system consisted of parallel processing system and an interface between a Macintosh computer acting as a front end processor(FEP) and a processor. The design is done in a way that the whole process can be expanded to 16 backplanes and each backplane consisting of 16 processing elements as shown in Figure 3-13[15]. A commercially available Vesta SBC332 Board is used as the basic processing element, each dedicated to control a specific subsystem of the whole robot. Each board contains a Motorola MC68332 microcontroller and onboard RAM and EPROM up to 1 Mbyte each. Those independently controlled processors communicate through dual port RAMs(DPRAMs), which allow two processors to share the memory space within it, permitting information exchange with other processors to complete tasks such as hand-eye coordination. Viewing from this point, the MC6811 motorboard acts as a slave of this system. The FEP is interfaced entailing the use of a Motorola MC68332 to act as an intermediate front end processor(InterFEP). FEP and InterFEP are interfaced with a SCSI bus and InterFEP and the backplanes are interfaced through a serial port. The programming environment is based on the Macintosh and in particular runs in Macintosh Common Lisp. L, developed by Brooks[4] is a downwardly compatible subset of Common Lisp and it is run on each MIMD machine node. L is used to program the high level learning routines that are introduced in the next chapter.

3.3 Sensors

3.3.1 Exteroceptors

Manipulation learning does not occur without fully utilizing exteroceptor and proprioceptor sensory feedback. As exteroceptors, force sensing resistor(FSR) devices which resemble membrane switches are used. The sensors are less than 0.15 mm thick film that are wrapped around the surface of the fingers and the palm. The construction of the sensor is based on two polymer films of sheets as shown in Figure 3-14. A conducting pattern is deposited on one polymer in the form of a set of interdigitating electrodes and a proprietary semiconductive polymer is deposited on the other sheet. The sheets are faced and laminated together with a combination adhesive spacer material. With no applied force, the resistance between the electrodes is high, and the resistance drops as the force increases, following a power law relationship. Two 2 inch $\times$ 2 inch square FSRs are wrapped around each phalange. For the palm, four position sensing resistors(PSRs) and a large FSR which covers the entire palm are used. A linear potentiometer, a kind of PSR shown in Figure 3-15, measures the position of an applied force along its sensing strip. A voltage, generally 5 volts, is applied

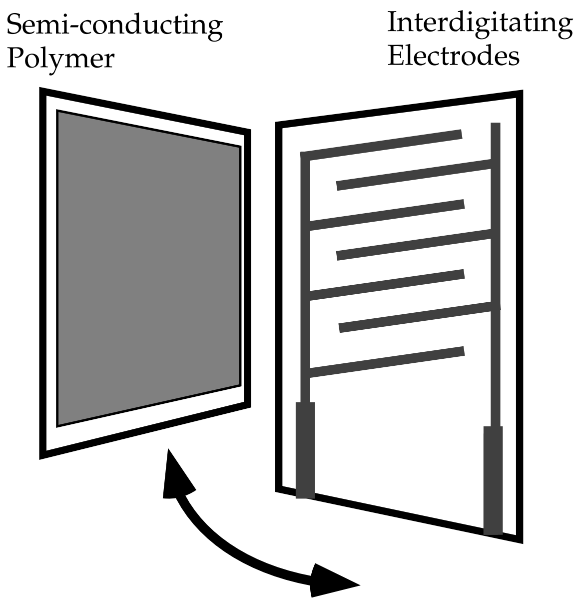

Figure 3-14: Commercial Force Sensing Resistor structure.

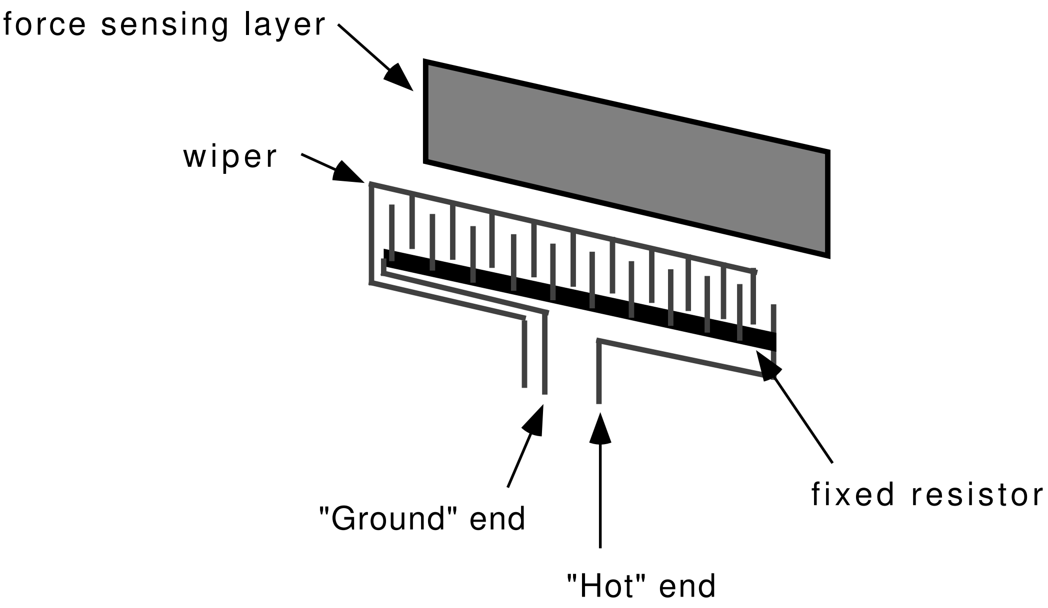

Figure 3-15: Commercial Position Sensing Resistor structure.

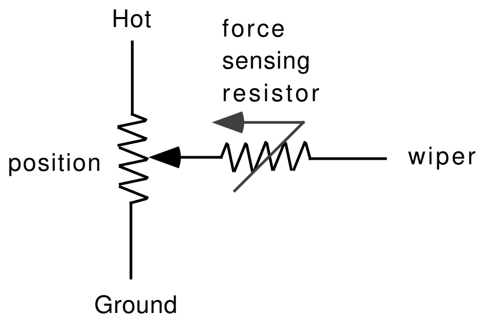

Figure 3-16: PSR equivalent circuit.

between the Hot and Ground ends of the fixed resistor strip. When force is applied to the force sensing layer, the wiper contacts are shunted through that layer to one of the conducting fingers of the resistor strip. The voltage read from the wiper is thus proportional to the distance along the strip that the force is applied. An equivalent circuit for this arrangement is shown in Figure 3-16. Position sensing resolution can be approximated by

$$\Delta x = \frac{2w_s^2}{w_f} \tag{3.20}$$

where w_s is the width of the conductive fingers, normally 0.5 mm, and w_f is the width of the applied force with an assumption of a constant force across the force footprint. One drawback of this material is that the force measurement is of one point only. If multiple locations are stimulated, the barycentric position, a positional average weighted over the force distribution,

$$x_{ave} = \frac{\int_{ground}^{hot} xF(x)dx}{\int_{ground}^{hot} F(x)dx} \tag{3.21}$$

where x is the positions of contact and $F(x)$ is the force distribution, is measured.

Since these measurements are processed at a sensor interface board on the dorsum, wires must go through the inside of the phalanges and the constantly moving joints. To accommodate the situation, the sensor is modified by eliminating an interface strip and attaching commercially available durable and flexible wires to the surface of the

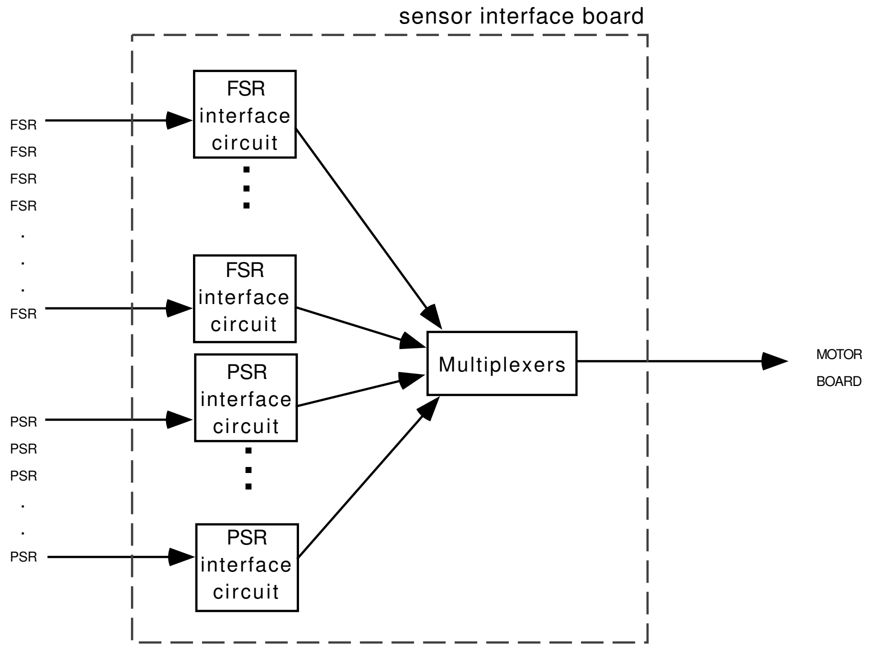

Figure 3-17: A block diagram of a sensor interface board.

film using a conductive adhesive epoxy. The epoxy is chosen so that it solidifies in room temperature, avoiding to melt the film of the sensor, and the hardness value is low when solidifies. All the wires are connected to a sensor interface board where the sensed resistance values, R_{FSR}s and R_{PSR}s, are processed. The block diagram of the interface board is shown in Figure 3-17. The FSR signals are interfaced using a simple force to voltage conversion as shown in Figure 3-18. The output is described by the equation,

$$V_{out} = \frac{V_+}{1 + R_{FSR}/R_M} \tag{3.22}$$

where V_{out} is the output voltage, V_+ is the supply voltage, and R_M is the measuring resistor value. According to the equation, the voltage output increases proportional to increasing force. R_M is chosen to maximize the desired force sensitivity range and 4.7 KΩ is used for the hand. For the PSRs, an output can be read through a simple voltage follower as a buffer as shown in Figure 3-19. In order to prevent the high current from flowing through the sensor during the measurement, it is important to

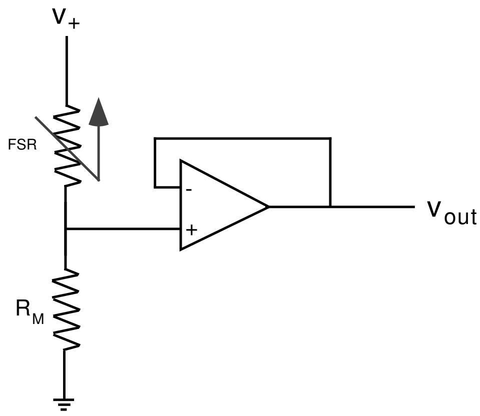

Figure 3-18: A FSR interface: force to voltage conversion circuit.

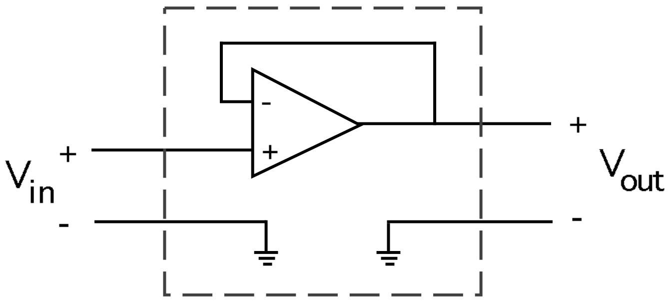

Figure 3-19: A PSR interface: A voltage follower.

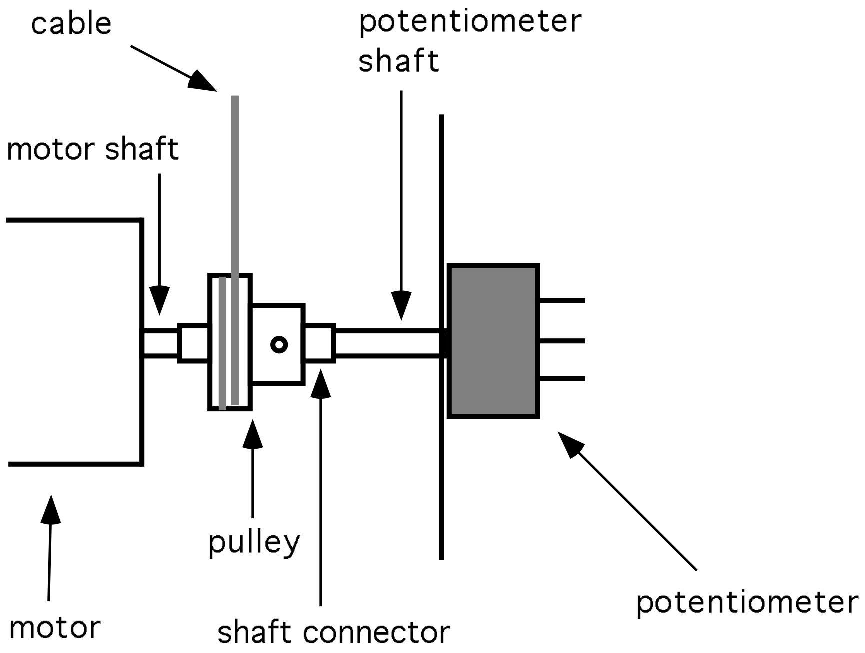

Figure 3-20: A rotational potentiometer measuring the position of a motor.

incoporate a buffer because a low-resistance load is driven by a source with a high resistance requiring isolation.

3.3.2 Proprioceptors

Proprioceptors respond to changes in the position of the body or its parts. Fundamentally, the use of motors is not anthropomorphical since joints for human are not controlled by rotational forces, but a force applied by muscles. Muscles receive an abundant supply of nerve endings acting as proprioceptors, while their functionality is still not very clear. Since muscles are still impossible to implement in the way our muscles work, the usage of actuators is not avoidable. With an implementation using motors, it is possible to measure the rotational position of the motors. To minimize size and weight, rotational potentiometers are used instead of optical encoders as shown in Figure 3-20. The information gathered is filtered through an RC circuit and processed to an 8 bit digital signal at a MC6811 analog-to-digital converter port.

This gives a 2^8 resolution per a rotation of a motor, which accounts to $180°$ resolution between curled and expanding configuration of a finger, which is much more precise than what humans are capable of measuring without visual feedback.

Another proprioceptor used is a current sensor that is a built in capability of a L293E motor driver. A load current , which can be as high as two volts, is converted to voltage information with a resistor avoiding a high current flow to the microcontroller. This information both protects the motor from overheating, and permits a measurement of how hard a finger is at work at each instance of a grip.

Chapter 4

Learning Process

Learning, storing past experience in the brain to guide future action, is an effective way of refining hand movements. In the early 20th century, Ivan Pavlov argued that conditioned reflexes form a basis for all learned behavior. In the 1930's Burrhus F. Skinner argued that only outcomes such as rewards and punishments caused learning, though many psychologists argued against it. As of today, the nature of learning is still not clear.

This chapter presents two nervous systems that have been implemented for this thesis. One is a system that is normally controlled at the spinal cord level such as reflex, and the other is a higher level learning system that utilizes neural network theory. The overall nervous control system is shown in Figure 4-1.

4.1 Low Level Controller

The low level calculations are all done in the MC6811 mounted on the palm and programmed using Assembly language.

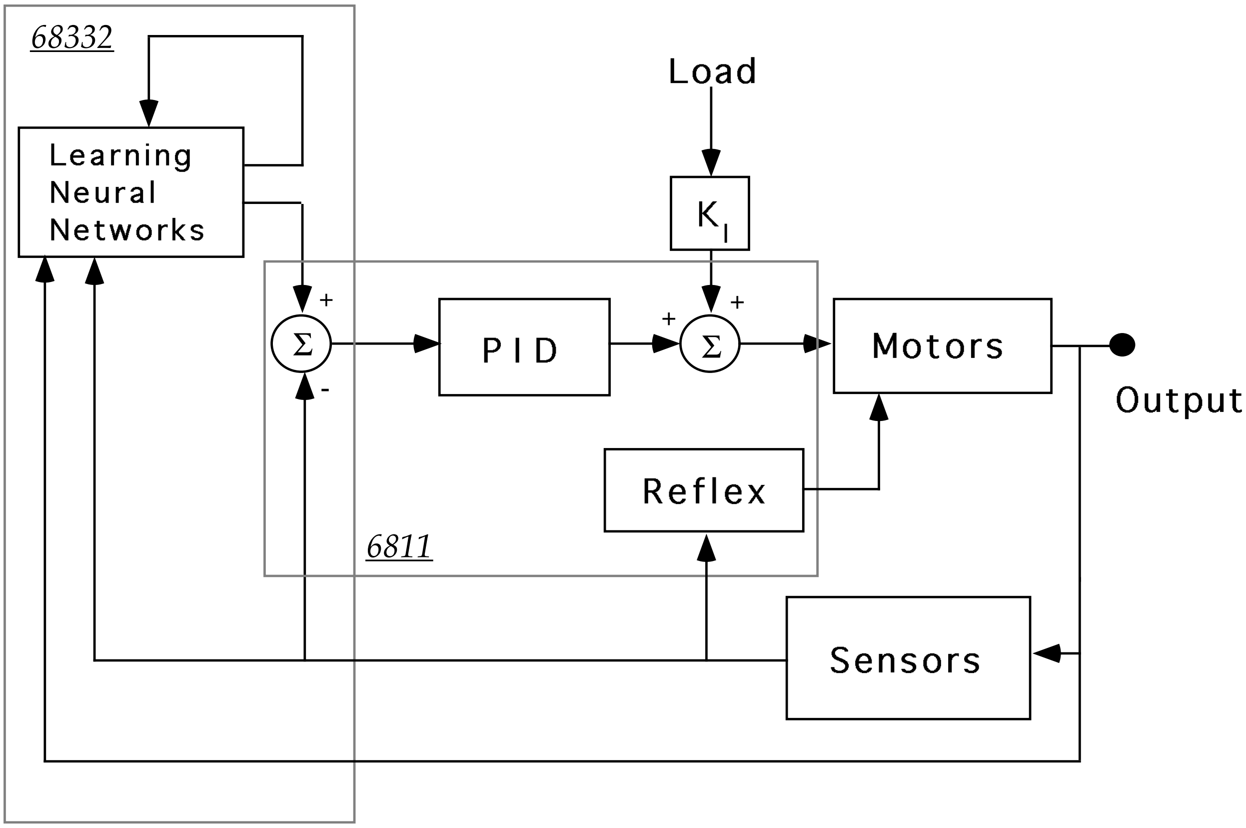

Figure 4-1: A block diagram of overall nervous control system implemented.

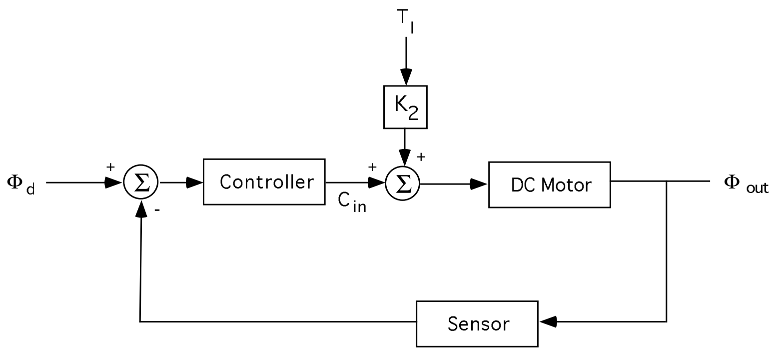

Figure 4-2: A simple block diagram of feedback control system.

4.1.1 PID Control

The control structure for the finger movement needs to be a closed loop to compensate for noise from the environment and to let the system converge at all times. The dynamics of a DC motor in a control loop shown in Figure 4-2 can be expressed as

$$\tau\ddot{\Phi} + K_0\dot{\Phi} + \Phi = K_1(C_{in} + K_2 T_l) \tag{4.1}$$

where τ is the time constant, Φ is the motor rotational position, C_{in} is the input from a controller, T_l is the load torque, and the K_m's are constants related to the motor characteristics. τ is determined by the characteristics of the motor and when it becomes smaller, the closed loop system becomes faster and more desirable. From Equation 4.1, using Laplace Transform, the motor process can be written as

$$\text{Motor} = \frac{K_1}{\tau s^2 + K_0 s + 1}. \tag{4.2}$$

For this system, a proportional plus integral plus derivative(PID) controller is chosen because of its ability to provide an acceptable degree of error reduction while simultaneously providing sufficient stability and damping[9]. For this system, the controller can be written as

$$\text{Controller} = G(1 + \frac{1}{T_I s} + T_D s)E \tag{4.3}$$

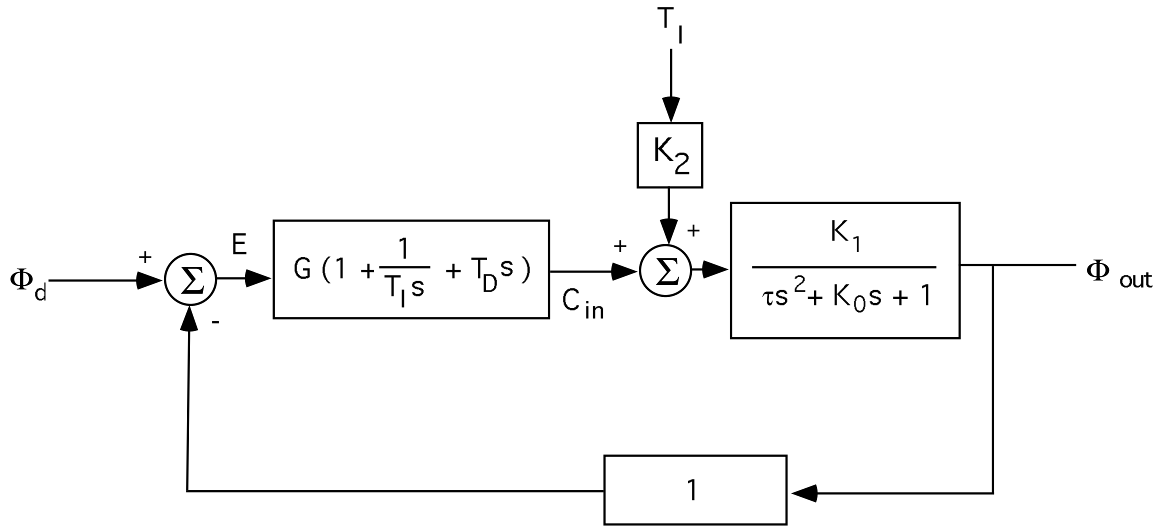

Figure 4-3: A block diagram of finger position control system.

where G is the feedback gain, T_I is constant called integral time, T_D is a constant called derivative time, and E is the error. For the sensor, a potentiometer reading Φ is used, so the gain for the sensor is 1. The system described above is shown in Figure 4-3, and the output is calculated to be

$$\Phi = \frac{K_1 EGT_D T_I s^2 + (EGT_I + K_2 T_I T_l / EG)s + EG}{T_I s(\tau s^2 + K_0 s + 1)} \tag{4.4}$$

where E is $\Delta\Phi$ containing no s term. Therefore, this system converges with time at all time.

4.1.2 Reflex

Reflex is a system that is controlled at the spinal cord. A curling reflex, only exists for infants, and allows the fingers to curl when the inner surface of palm is touched. A releasing reflex reaction occurs when an intolerable amount of stimulus is applied to the skin. A releasing reflex is useful for both avoiding the physical damage of the hand and to learn the limit of its capability. A curling reflex is important at the learning stage, but it can be eliminated eventually. Based on these ideas a very simple reflex system is implemented using force sensing resistor(FSR) sensory feedback. When an FSR senses a signal higher than its threshold resistance, the joints are commanded in a way that the finger moves in the opposite direction from where the stimulus is applied. The normal command sent to motors are overwritten by the reflex signals. If the inner skin is weakly stimulated, all the fingers are commanded to curl until the sensor reading reaches 30. This pressure is not strong enough to hold an object, but it simulates babies' reflex systems. This curling reflex initiates the learning process described in the next section.

4.2 High Level Neural Networks

Due to the lack of visual and auditory feedback, only the primitive learning processes that occur locally for the hand are considered in this thesis. For infants, different learning processes occur interactively and simultaneously. For example, think of a situation where an infant tries to lift an object off the ground, grasping, lifting the hand, and failing to lift up the object. From visual feedback, the infant recognizes that the object has slipped off the hand. By repeating this process, they learn to connect the visual "slip" with their sensory information. Adults can apply the right amount of force to hold an object by applying enough by not excessive force to an object without slipping. This operation is possible due to repeated practice at the initial grasping learning stage. Simultaneously, when the infant touches and drops the object, joint proprioceptors and exteroceptors on the skin react in a certain way. After some repetitions, the infants connect the relationship of sensory information with objects' hardness, texture and weight. All those separate learning processes merge to create our consistent stable manipulation skills. For this thesis I implemented three learning processes separately, each utilizing neural networks using different strategies. First, object hardness recognition learning is conducted using a competitive learning strategy. Second, a three layered backpropagation algorithm is used to train the shear detection. By applying those two trained networks, the optimal grasping action is searched by a reinforcement learning strategy (that is somewhat similar to Q-learning technique).

4.2.1 Hardness Recognition Network

Theory of Competitive Learning

Topologically, there is substantial evidence for the spatial self-organization of brain areas that contain sensory or motor maps [8]. For some stimuli, there is some form of competition between activities of neurons on the neural surface. The idea of competitive learning was originally proposed by Rosenblatt [29], and implemented by many [17] successfully. Competitive learning contains lateral feedback, which depends on the lateral distance from the point of its application. From biological inspiration, lateral feedback is described by a Mexican hat function, shown in Figure 4-4. A short

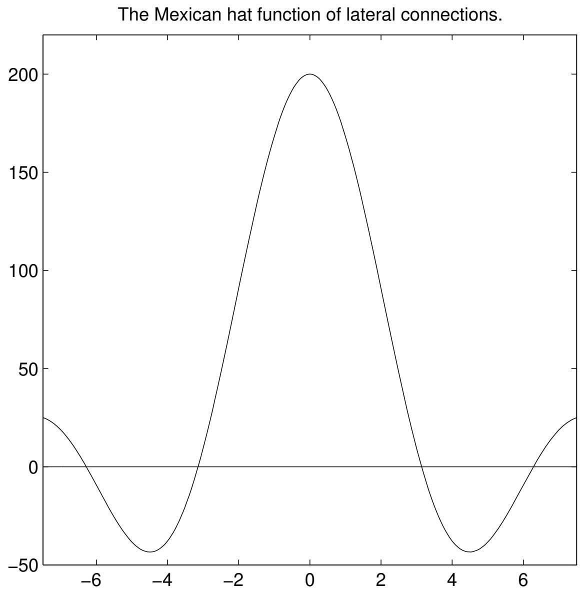

Figure 4-4: The Mexican hat function of competitive learning lateral connections.

range lateral feedback has an excitatory effect and a penumbra lateral feedback has an inhibitory effect. The output signal of neuron i, y_i, at time step $n+1$ can be expressed in a following difference equation:

$$y_i(n+1) = \psi(\sum_{j=1}^{p} w_{ij}x_j + \beta \sum_{k=-K}^{K} c_{ik}y_{i+k}(n)), \quad \text{for} i = 1, 2, ..., N \tag{4.5}$$

where $\psi(f)$ is some nonlinear function to ensure $y_i \geq 0$, w_{ij} is the synaptic weight of jth feedforward connection, p is the number of input terminals, x_j is the jth input signal, β is the feedback factor that controls the rate of convergence of the relaxation process, K is the radius of the lateral interaction, c_{ik} is the lateral feedback weight connected to neuron i, and N is the number of neurons in the network.

Application

Utilizing competitive learning theory, the hardness of objects can be categorized over time. The experiment is conducted with eight different objects of same size and different compressibilities. Each object is touched by curling one finger around the object very slowly. Precisely taking three seconds to fold fingers fully, hold for two seconds, and straighten the finger taking three seconds. The sensory readings are taken from both force sensors on the finger and the potentiometer reading of the motor controlling the finger which are converted to an eight bit digital information. The program is written in 6811 in a way that the readings are recorded every 0.14 seconds. The raw data extracted from a finger is shown in Figure 4-5. The potentiometer reading, $p(t)$, indicates the position of the finger. The derivative of $p(t)$ has three distinct characteristics.

$$\frac{dp(t)}{dt} = \begin{cases} C_1 & C_i = constant \neq 0 \\ -C_2(t-t_1) + C_3 & \\ 0 & \end{cases} \tag{4.6}$$

As the finger curls, the motor moves at the constant rate when the finger does not contact the object surface. At this stage, $dp(t)/dt$ is a non zero constant. When

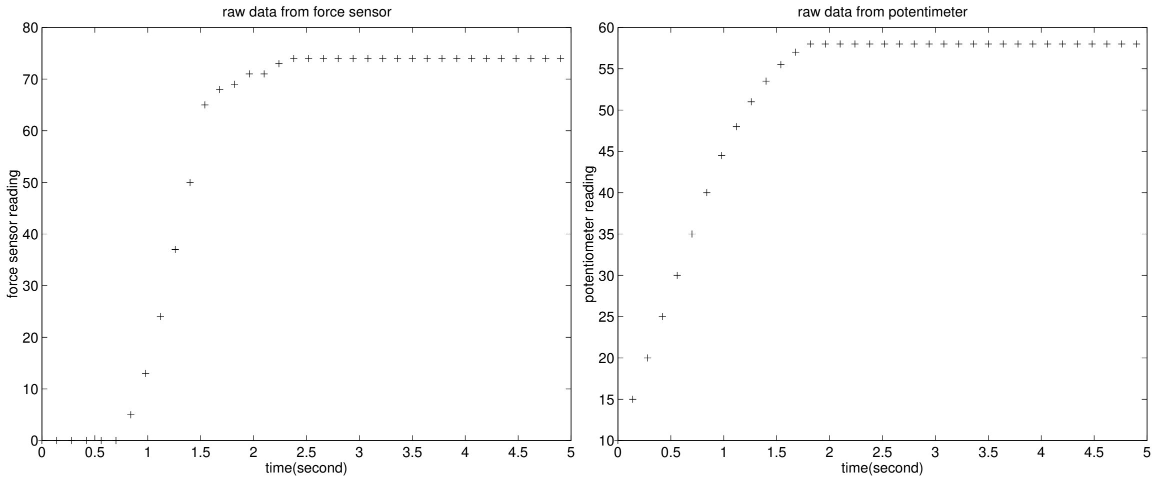

Figure 4-5: Hardness recognition raw data extracted from a meduim hardness object during the finger folding stage.

the object is firmly grasped, the finger stops curling and results in $dp(t)/dt \rightarrow 0$. The significant difference between different hardness object can be seen in the stage where $dp(t)/dt$ is not constant. This stage signifies that the object and the finger are in contact, but the object's compliancy is letting the finger continue to move. Objects have a constant compliance factor, C_2, which is proportional to the hardness of the object. The comparison of two objects with different hardness are shown in Figure 4-6. It seems as if the hardness of objects can be categorized using only this information. However, repeated experiments with the finger shows some unexpected results which may not be relevant to humans because of our superior tactile sensory system. Due to the nature of the force sensors, they are not capable of sensing a force smaller than 20 grams. When a very spongy object is grasped, the sensor cannot detect the contact until the object is squashed enough to give some force back to the finger. Therefore, a very spongy object gives a similar response as a hard object. One sensory difference in those two objects is the force reading when the object is completely compressed and held. Since the spongy object has the resistance force orthogonal to the finger surface, the force reading is much higher than for the harder object. These analysis show why both potentiometer and force sensor information

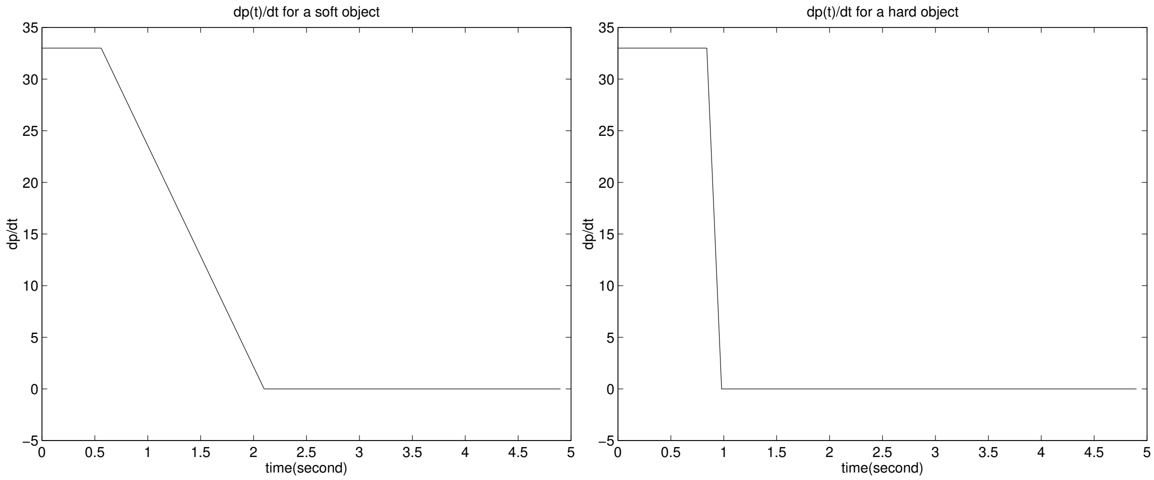

Figure 4-6: $\frac{dp(t)}{dt}$ comparison for soft and hard objects.

are crucial in distinguishing the object hardness.

Experimental Results

For each curling experiment, two numbers are extracted and recorded. The first is the duration of $dp(t)/dt$ non constant time, Δt. It is expressed in digital units where one unit is 0.14 seconds. The other is the maximum force sensor reading expressed in a seven bit digital number. Eight bit information is shifted one to the right to eliminate small noise. Eight different objects are tested ten times each and the results are plotted in Figure 4-7. Using those data as inputs, a 2 layer, 6 neuron competitive network is constructed with random initial synaptic weights and trained. Figure 4-8 shows the trained neurons over the input map as they get trained. Since this is unsupervised learning, the initial randomness can confuse the neurons to categorize somewhat different from what was intended when the training session is too short or the learning rate is too high(a confused neuron is shown in Figure 4-9). Even with bad initial random weights, such as the one causing the confused neuron, the result converges after 500 epochs. Once the network is trained, different inputs can be fed to the network to find the category of the touched object. This strategy works well for this purpose since there is no clear cut way to categorize objects. The trained

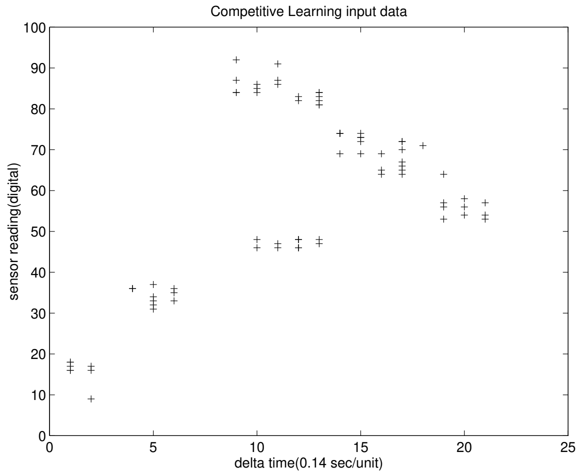

Figure 4-7: Competitive learning input data.

network was tested with data taken from objects not used for training and shown in Figure 4-10. With very diverse test objects, the sensory readings fell closely to the trained neurons. Initially training the network with six diverse hardness categories gives a good distribution of graspable objects. Even if an object with dramatically different compliancy is found, it only takes roughly 10 experiments to take data and 500 epochs to retrain, all of which takes less than one minute to do.

4.2.2 Shear Detection Network

Theory of Back-Propagation Algorithm

The back-propagation algorithm is the most popular application of multilayer perceptrons for supervised learning. The process consists of a forward pass and a backward pass with known desired output signals $d(n)$ where n is the instance of the number of training. The inputs is applied to the forward pass network and fed through layer by layer. The net internal activity level $v_i^{(l)}(n)$ for neuron i in layer l is

$$v_i^{(l)}(n) = \sum_{j=0}^{p} w_{ij}^{(l)}(n) y_j^{(l-1)}(n) \tag{4.7}$$

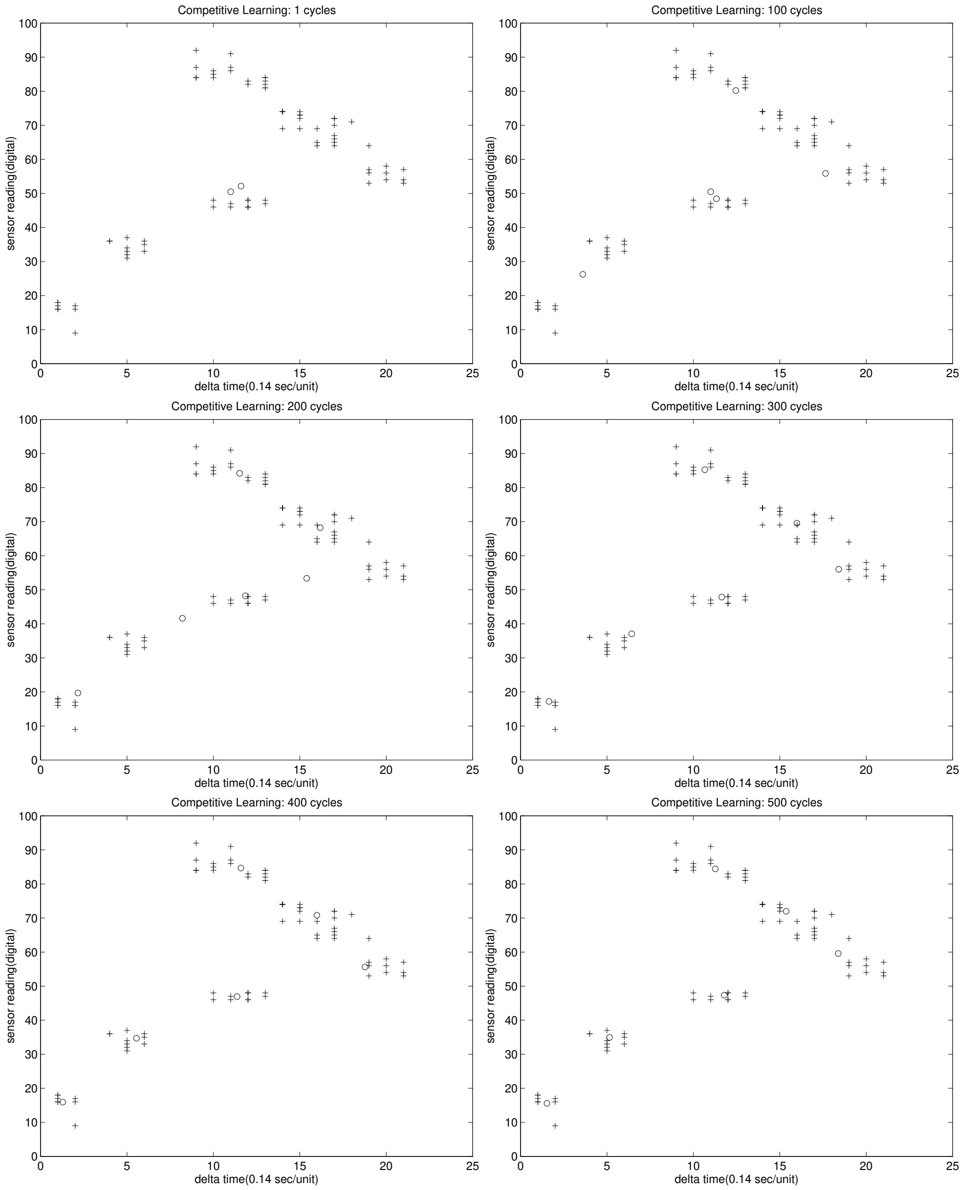

Figure 4-8: Hardness recognition competitive learning training steps: '+' are inputs and 'o' are the neurons.

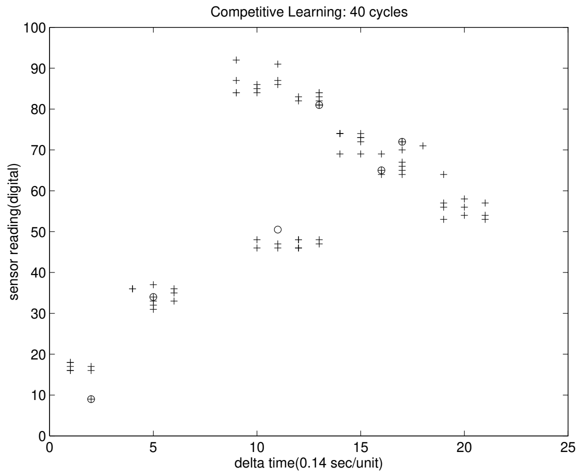

Figure 4-9: Competitive learning containing confused neuron(the two neurons around (17,70) should be spread apart to the other input cluster around (20, 60)):
'+' are inputs and 'o' are the neurons.

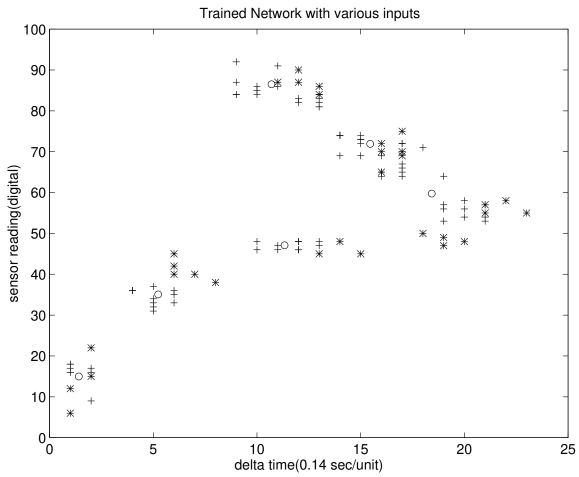

Figure 4-10: Hardness Recognition: Competitive learning trained network with testing inputs(all the test inputs are clustered around the existing trained neurons)

where $w_{ij}^{(l)}(n)$ is the synaptic weight of neuron i in the layer l that is fed from neuron j in layer $l-1$ at iteration n and $y_j^{l-1}(n)$ is the function signal of neuron j in the layer $l-1$. At the output of each neuron in all the layers there is nonlinear smoothing function, a sigmoid,

$$y_i^{(l)}(n) = \frac{1}{1 + exp(-v_i^{(l)}(n))} \tag{4.8}$$

to make the function differentiable. At the output layer, L, the set of outputs is compared to the desired value giving an error signal,

$$e_i(n) = d_i(n) - y_i^{(L)}(n) \tag{4.9}$$

which is propagated backward layer by layer against the direction of synaptic connections adjusting the synaptic weights in the following manner:

$$w_{ij}^{(l)}(n+1) = w_{ij}^{(l)}(n) + \alpha(w_{ij}^{(l)}(n) - w_{ij}^{(l)}(n-1)) + \eta\delta_i^{(l)}(n)y_j^{(l-1)}(n) \tag{4.10}$$

where η is the learning rate, α is the momentum constant, and the local gradient, δ is

$$\delta_i^{(L)}(n) = e_i(n)y_i^{(L)}(n)(1 - y_i^{(L)}(n)) \tag{4.11}$$

$$\delta_i^{(l)}(n) = y_i^{(l)}(n)(1 - y_i^{(l)}(n))\sum_k \delta_k^{(l+1)}(n)w_{ki}^{(l+1)}(n). \tag{4.12}$$

The algorithm is to iterate these computations until the network stablizes within the bounds of targeted error.

Application

Visually, it is obvious when an object slips from a hand. From repeated shear experience, the relationship between the sensory information on the fingers and the result develops for infants. Shear is locally detectable sensory information if there exist multiple rows of pressure sensors perpendicular to the direction of slip. With the way the robot hand for this research is oriented, the palm is perpendicular to the ground and fingers are horizontal, which makes the three fingers orthogonal to the direction of slip. In order to simulate the shear learning process, sensory data from the fingers are used as inputs and the visual feedback about the existence of shear is used as the desired output to train a feedforward network. Since shear is a time dependent process, the input signals have to contain multiple time space sensor readings. The size of the input signal vector is defined as

$$(row, col) = (tf, m^{tf}) \tag{4.13}$$

where t is the number of discrete time steps, f is the number of finger sensors used and m is the number of sensory reading levels. This size needs to be minimized in order to speed up the learning operation. Straight out of the microcontroller, there are $m = 2^7$ sensory reading levels. Obviously seen from equation 4.13, it will take all day to just feed forward an input of this size. Also for a noisy environment, this is not an optimal implementation. As a solution, m is reduced to two numbers, 5 and 0, as the maximum and the minimum inputs. Back-propagation classifier can generalize the numbers between maximum and minimum well with an optimal number of layers and without overtraining. When the data is overtrained, the inputs are overfitted and cannot adapt the values between 4 and 1. Reducing m to two still contains enough information conserving the physics of shear and makes the calculation much simpler and faster. Since slipping is not a reversible operation without an external force applied, recording two discrete time steps with an optimal step size is satisfactory. If the step is too small, most of the calculation will be wasted detecting no changes in the readings. However, if the step is too large, the shear will not be detected quickly enough. To calculate the maximum speed of object slipping, assuming no friction, the equation

$$x - x_0 = v_0 t + \frac{1}{2} a t^2 \tag{4.14}$$

can be used. Since $v_0 = 0$, time it takes for a point of an object to slip from one finger to the other is 0.06 seconds. Therefore a step size of 0.28 seconds is chosen. With those assumptions, the input vector has the size of (6, 64). When the columns of the vector are examined, there are 10 columns of inputs that are not realistic or are ambiguous so the size can be reduced even more to (6, 54). The desired output data is one bit information, 1 being shear detected and 0 being no shear.

Experimental Results

Having six input nodes and one output neuron, a four layer with two hidden layer feedforward network is constructed. Because of the simplification made for the sensory inputs, by rounding up the data and reducing the m to smaller numbers as following,

$$\begin{cases} 81 \sim 127 & \rightarrow 5 \\ 61 \sim 80 & \rightarrow 4 \\ 41 \sim 60 & \rightarrow 3 \\ 21 \sim 40 & \rightarrow 2 \\ 1 \sim 20 & \rightarrow 1 \\ 0 & \rightarrow 0 \end{cases} \tag{4.15}$$

a four layer network was found most optimal for the generalization to occur well. The inputs are taken from the sensors on the three fingers as the fingers curled around the given object, a paper cup. Since the hardware is not ready to run the hand completely autonomously, some external force was applied to reach the grasping figure. When slip is not detected the computer is given a default signal 0 which signifies the non-slip stage. When it is detected, a 1 is manually typed in through the serial port as a visual feedback signal overwriting the default input. Again, since the visual system

network	*learning rate*
2 neuron hidden layer network	1.58
6 neuron hidden layer network	1.01
15 neuron hidden layer netowork	0.65

Table 4.1: The optimal learning rate for each network.

is not at the stage where it can cooperate with the hand, the experimenter has input the signal when the visually obvious slip is detected. After enough cases of slip were introduced, all the sensory data was recorded and the network was trained separately from the hardware. Eventually I would like to train the network on line, but without having the real visual feedback, manual labor is overwhelming. To record one set of input vector to run one epoch, about 50 different slips are manually inputed. And to train the networks, at least 500 epochs are required. In the training session, the number of hidden layer neurons and the learning rate were varied to find the optimal back-propagated networks. The number of neurons in the first hidden layer was fixed to 6 to match the number of input nodes. The number of neurons in the second hidden layer is deviated to 2, 6 and 15 neurons. Setting the desired sum-squared network error,

$$E(n) = \frac{1}{2} \sum_{i \in C} e_i^2(n) \tag{4.16}$$

to 0.006, I have trained the networks with different learning rates. If the desired error was not reached within 500 epochs, the training was stopped. The results are graphed and shown in Figure 4-11, Figure 4-12 and Figure 4-13. Since the initial random weights give different sum-squared error initially, comparing the speed of convergence between different networks is not relevant. Intuitively, all the networks converges faster when the learning rate is increased. However, as soon as the learning rate exceeds the fastest convergent point, the systems never converge and seem to get stuck in a local minima at $E(n) = 19.00$. The optimal learning rate for each network is shown in Table 4.1. Even if the system converges at the end, the error does not stably decrease when the learning rate is higher. This makes the system unreliable

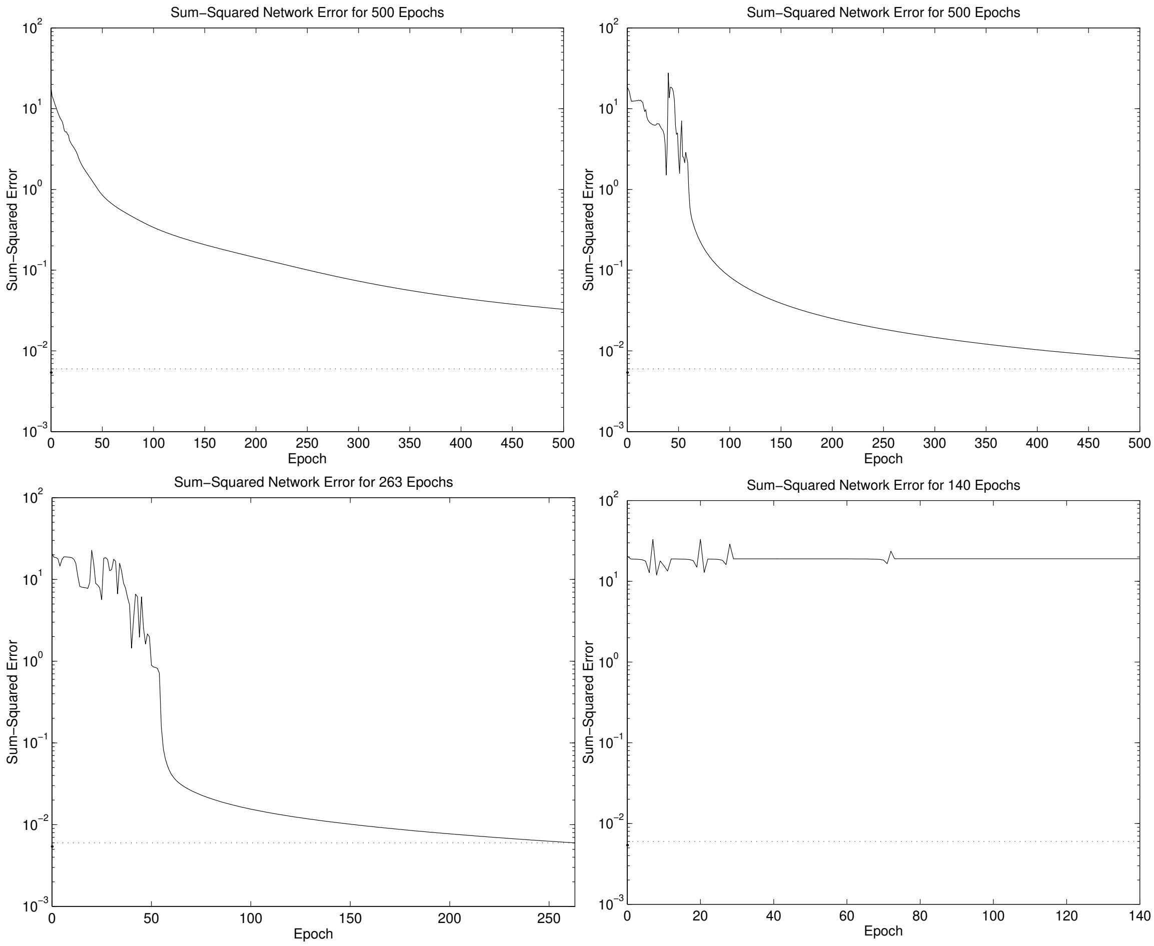

Figure 4-11: Two neuron hidden layer training result:
top left, $\eta = 0.1$;
top right, $\eta = 0.8$;
bottom left, $\eta = 1.5$;
bottom right, $\eta = 2.0$.

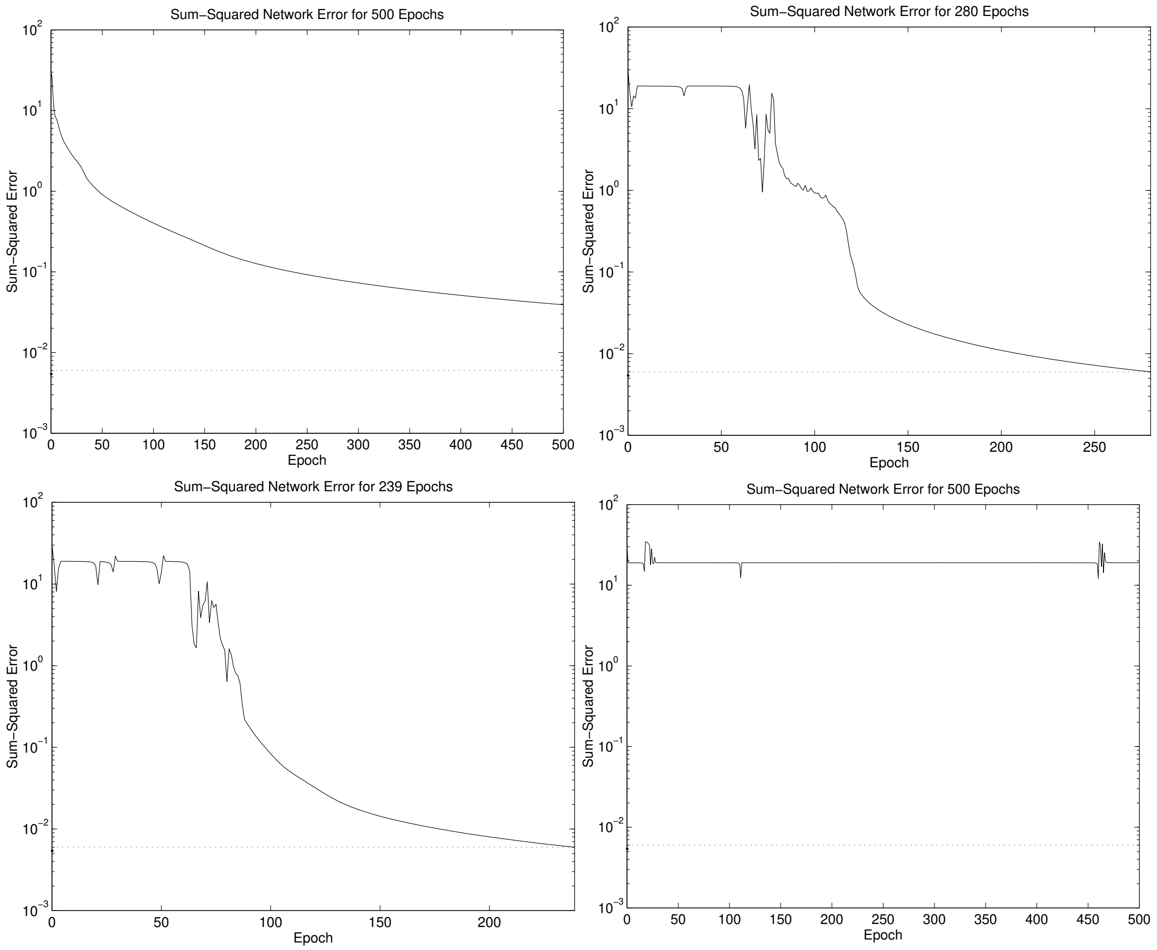

Figure 4-12: Six neuron hidden layer training result:
top left, $\eta = 0.1$;
top right, $\eta = 0.8$;
bottom left, $\eta = 1.01$;
bottom right, $\eta = 3.0$.

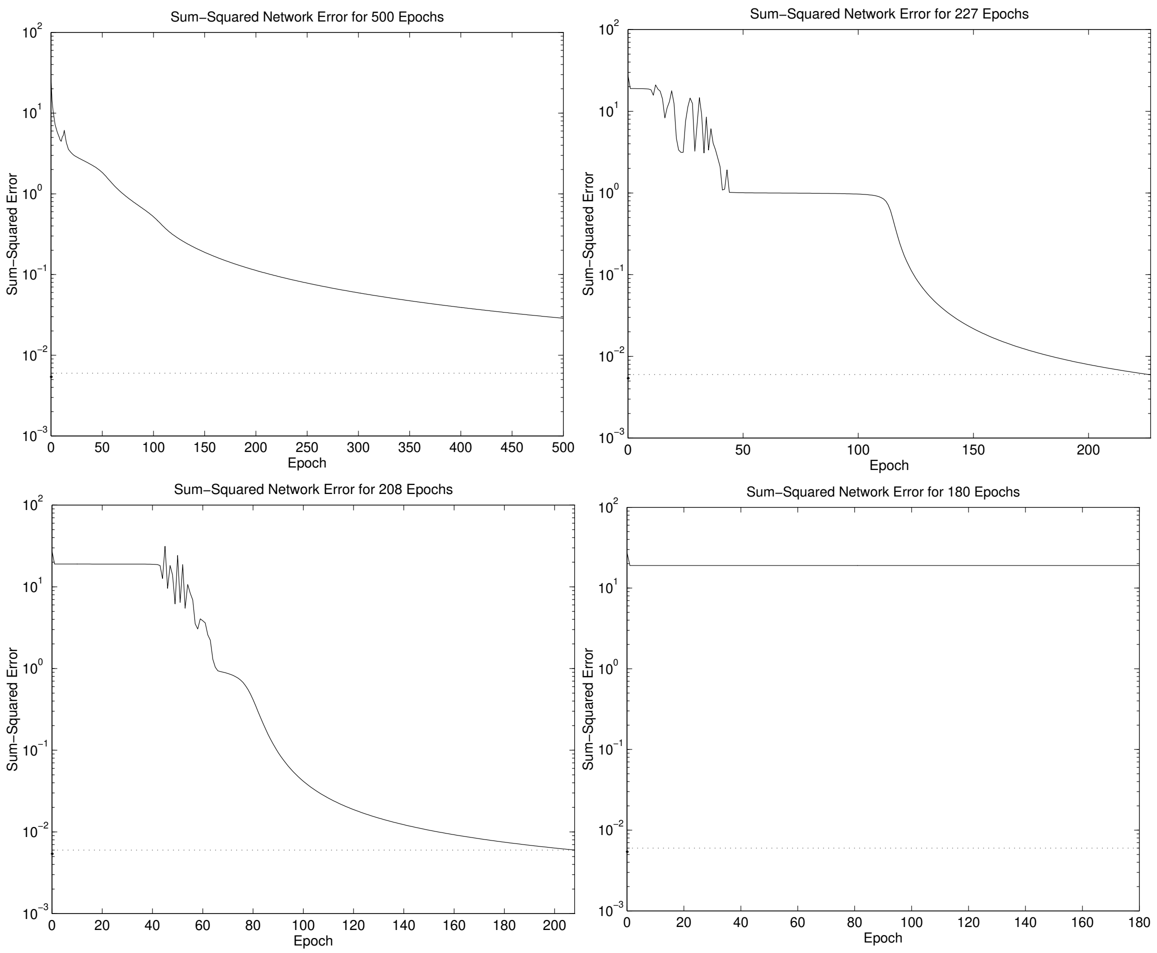

Figure 4-13: Fifteen neuron hidden layer training result:
top left, $\eta = 0.1$;
top right, $\eta = 0.5$;
bottom left, $\eta = 0.6$;
bottom right, $\eta = 0.8$.

since depending on the inputs, the system has a possibility of finding a local minima and never converging. Even though this problem may be solved when it is run on line with some noticable noise which can disturb away from local minima, it does make the system more reliable by picking a good middle ground learning rate. As far as the number of hidden neurons are concerned, the calculation time increases significantly as more neurons are added. Even though the network containing larger hidden layers can take higher learning rate stably, if each epoch takes longer to calculate, the advantage is diminished. For this specific experiment, six hidden neurons for both hidden layers and having 1.0 learning rate seems to be the most optimal solution, though this may change as the system is trained on line in the future. Average outputs of a trained network taken under many operations containing slips are shown in Table 4.2, where input difference is the most significant sensor reading difference

Input Difference	*Slipped?*	*Output*
5	yes	0.9863
4	yes	0.9863
3	yes	0.9867
2	yes	0.9905
1	yes	0.9903
0	no	0.0007
-1	no	0.0002
-2	no	0.0003
-3	no	0.0003
-4	no	0.0003
-5	no	0.0003

Table 4.2: Trained slip detection network output with testing inputs.

between two readings. The output is well categorized even for the inputs that are not used for training such as 1 to 4.

4.2.3 Grasping Action Network

Theory of Reinforcement Learning

Reinforcement learning is based on a common sense idea that if an action is followed by a satisfactory state of affairs, then the tendency to produce that action is strengthened[33]. This idea was initially studied in psychology by Pavlov in learning work with animals. In neural networks, the studies are focused on actor-critic learning algorithm or Q-learning, both based on the temporal difference method[33, 35, 37]. An actor-critic system has two subsystems, one is an evaluation network which estimates the long term utility for each state and the other is a policy network which learn to choose the optimal action in each state. A Q-learning system maintains estimates of utilities of all state-action pairs and utilizes them to select a suitable action. The object of Q-learning is to estimate a real-valued function, Q, of states and actions, where $Q(x, a)$ is the expected discounted sum of future reward for performing action a in state x and performing optimally thereafter. This relationship can be expressed as:

$$Q(x_n, a_n) = E\left\{r_n + \gamma Max(Q(x_{n+1}, y))\right\} \tag{4.17}$$

where r_n is an immediate reward at step n, γ is a discount factor, $0 \leq \gamma < 1$, and y is the next state. The estimation of Q, Q_{est} is updated at each time step,

$$Q_{est}(x_n, a_n) := Q_{est}(x_n, a_n) + \beta_n(r_n + \gamma Max(Q_{est}(x_{n+1}, y)) - Q_{est}(x_n, a_n)) \tag{4.18}$$

where β_n is a gain sequence, and all the estimation is maintained within the function. A gain sequence has a characteristic such that $0 < \beta_n < 1$, $\sum_{n=1}^{\infty} \beta_n = \infty$ and $\sum_{n=1}^{\infty} \beta_n^2 < \infty$. Q-learning has been proven to converge at all time[35].

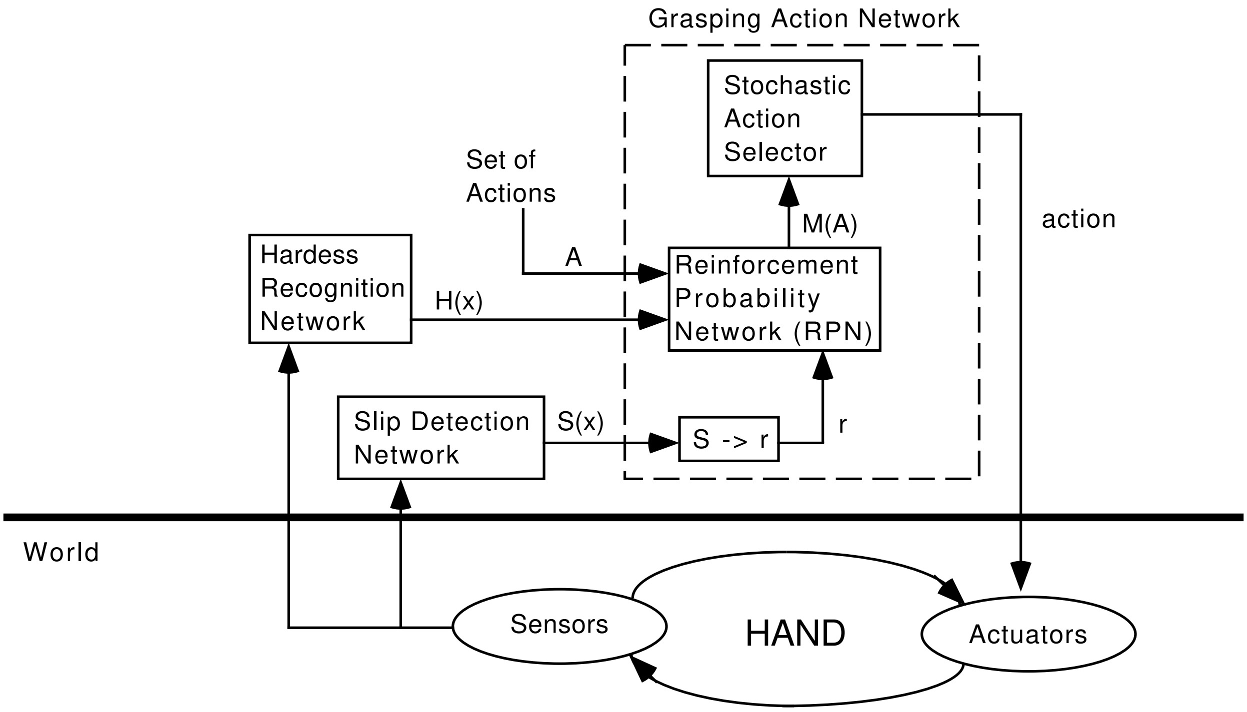

Figure 4-14: Grasping Action Network block diagram

Application

The Q-learning algorithm assumes that the system can observe an input vector at nth iteration, x_n, action chosen at Stochastic Action Selector at nth eteration, a_n, reinforcement value, r_n, and the next input vector, x_{n+1}, at each time step. However, since grasping is a one way operation(meaning *open* $\rightarrow$ *close*, not *open* $\leftrightarrow$ *close*), x_{n+1} cannot be seen at the end of the iteration, n. Moreover, x_n is already analyzed and categorized using competitive learning networks. Implementing with a connectionist idea, internal self reinforcement system was built using two components as shown in Figure 4-14. The first system is a Reinforced Probability Net, RPN, which takes the classified information, $H(x)$, from hardness recognition network and a set of actions, $A = \{a_1, a_2, ..., a_z \mid a(j) \equiv \text{a set of actuator inputs of jth action}\}$. It outputs an action merit vector, $M(A)$, that assigns a value to each action. The second system is Stochastic Action Selector that takes $M(A)$ and selects an action and sends the information to the actuators. According to the action given, the shear detection network gives an output which can be converted to an immidiate payoff value, r. RPN is reinforced using TD methods, back-propagating a reinforced correction vector, $RC(n)$. The simplified algorithm is as follows:

1. $H(x) \leftarrow$ current hardness class; for each action $a(j)$, $M(a(j)) \leftarrow RPN(H(x), a(j))$;
2. $a \leftarrow SAS(M(A))$;
3. Perform action a;
4. Send new sensory information to hardness recognition network and shear detection network; $(H(x), S(x)) \leftarrow$ new hardness class and shear value;
5. $r = -2S(x) + 1$;
6. $RC = M(A) + (\xi r)$; where ξ is a damping constant.
7. Adjust the RPN by back-propagating RC;
8. Go to 1;

There are two ways of implementing RPN. Classified RPN is shown in Figure 4-15. There are only two layers in the network with an additional neuron selector at the output. This allows the $M(A)$ to converge faster for each class, though when a new hardness category is added, it has to relearn by adding unattached neurons into the network and start from a scratch. The other implementation is Mutiple Layer RPN which uses more hidden layers and feed $H(x)$ with the action vector as shown in Figure 4-16. For this system, only synaptic weight adjustment is made for the existing neurons. This method varies in the time of retraining depending on the newly given object. For this experiment, the classified RPN is chosen to use due to the calculation speed and limited object hardness categories.

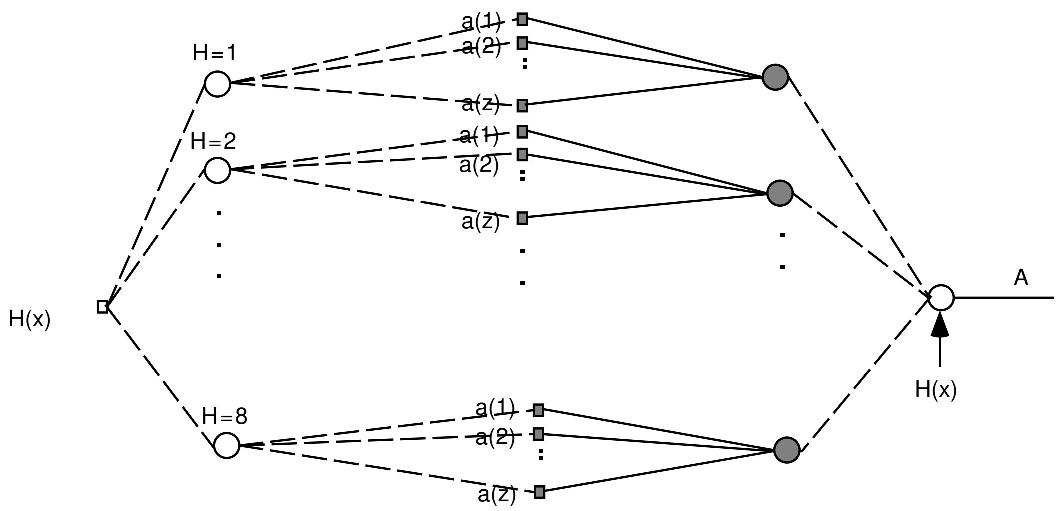

Figure 4-15: Classified RPN(Back-propagated on the solid lines)

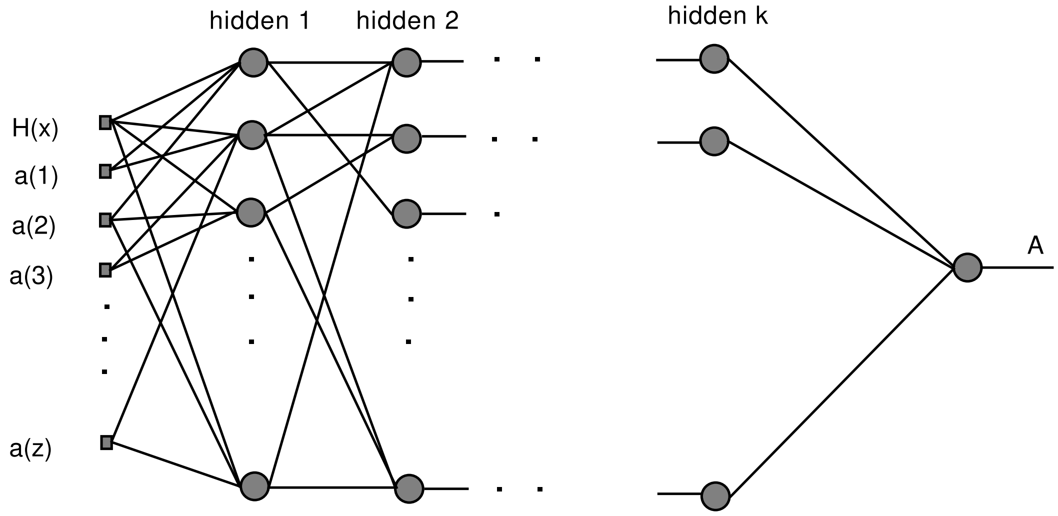

Figure 4-16: Multiple hidden layer RPN

Experimental Results

Six set classified RPN has been constructed with six categories from the hardness recognition network. Since each set gives a similar training result, only one class, $H(x) = 3$ is shown in this section. The set of actions has been determined to have eight cases of grasping potential positions. The initial weights are set in a way that each action has equal probability of being chosen at the beginning. It is a timely operation since, as mentioned before, the hardware is not functional enough to operate autonomously, when the grasping action signal is received, external force needs to be applied to achieve the grasping position and the slip is detected and input by the experimenter. For the Classified RPN method, the number of epochs can be quite small to achieve an optimally trained network. There are two variable constants, learning rate and damping constant, to change to achieve different ways of training the network. The learning graphs with different constant values in a short period of training are plotted in Figure 4-17 and the longevity training results are shown in Figure 4-18. When ξ is too small, the network never get trained as desired because the system is not reinforced strongly enough. Though as long as ξ is large, η does not need to be large to learn quickly and correctly. When both ξ and η are too large, the system falls into a local minimum and does not converge. The advantage of this system is that once the networks are trained within the desired square-sum errors, as long as the damping constant and learning rate are optimally small, the system can adapt to any new objects that are to be grasped. To simulate the trained network, the action chosen was output to a computer monitor through a serial port so that some external force can be applied to achieve the desired action. For a well trained network, 15 iterations were conducted and it chose one action that can achieve the stable grasp every time as shown in Table 4.3. If multiple actions can accomplish the grasp desired, the output actions are equally divided among them.

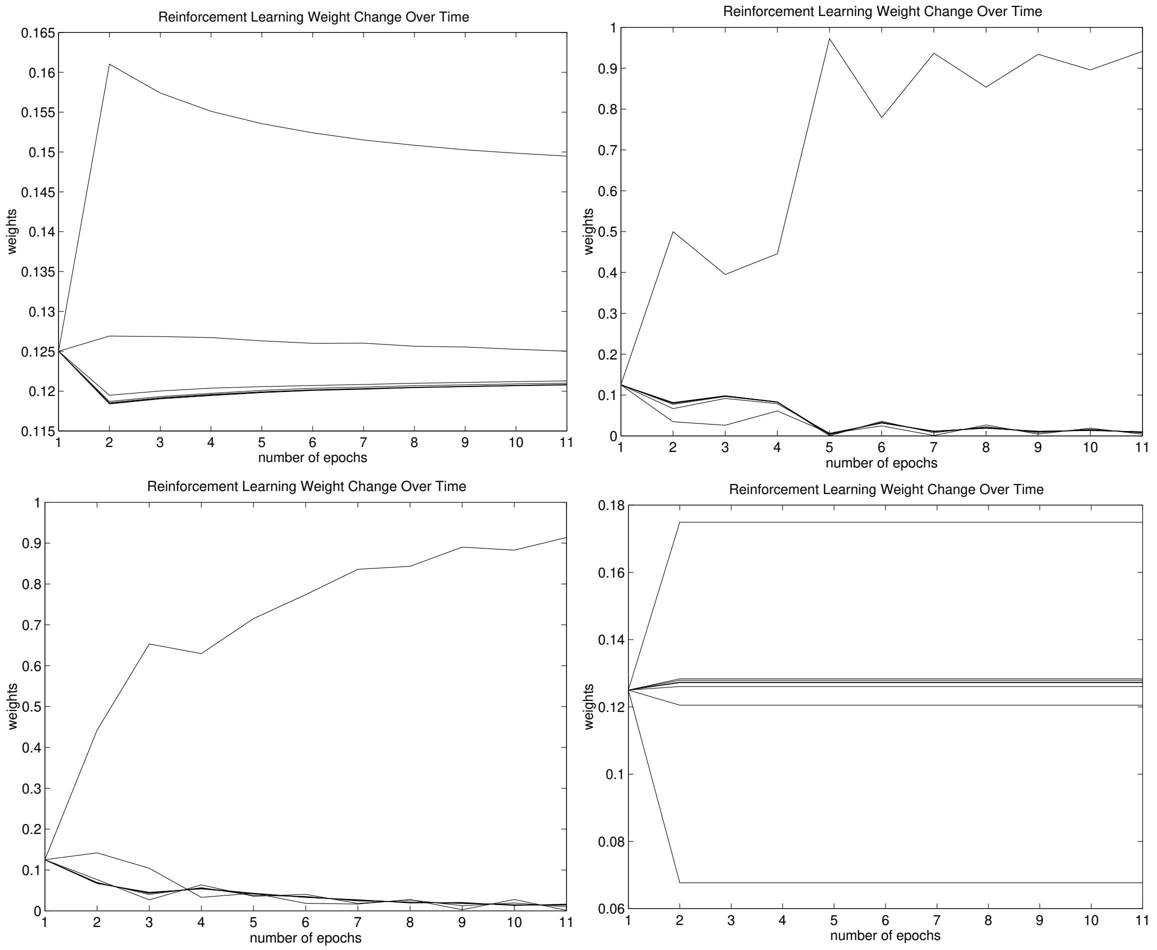

Figure 4-17: Changes in synaptic weights over short period of time for different learning rates and damping constant:
top left: $\xi = 0.1$, $\eta = 5$;
top right: $\xi = 0.9$, $\eta = 5$;
bottom left: $\xi = 5$, $\eta = 0.1$;
bottom right: $\xi = 5$, $\eta = 5$.

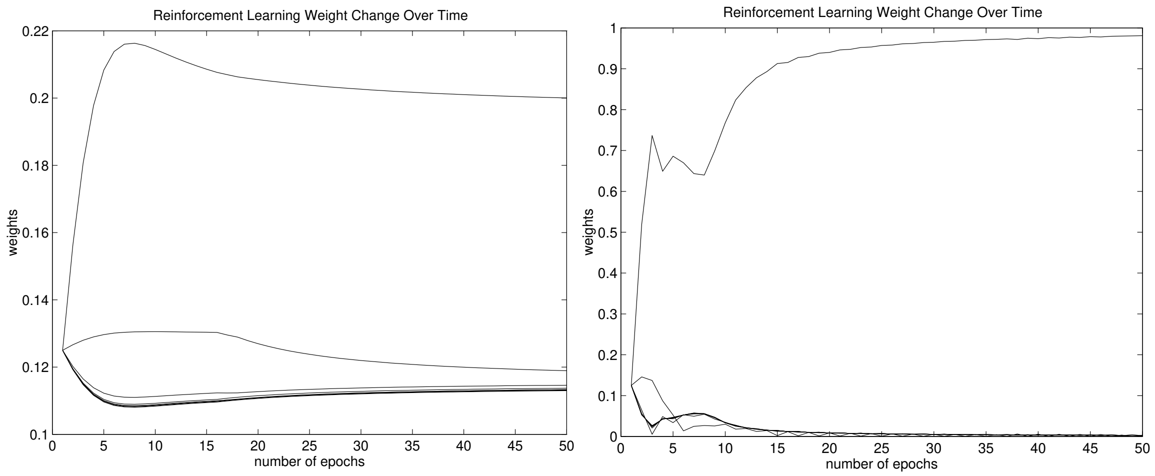

Figure 4-18: Training over a long period of time:
left: $\xi = 0.3$, $\eta = 0.3$;
right: $\xi = 1$, $\eta = 1$;

$H(x)$	$a(n)$	*Successful Grasping/# of trials total*
1	2	15/15
2	8	9/15
	2	6/15
3	6	15/15
4	1	7/15
	2	7/15
5	5	15/15
6	5	14/15

Table 4.3: Stable grasp success rate.

Chapter 5

Conclusions

5.1 Review of Thesis

For this thesis, a self-contained anthropomorphic scaled non-task driven tool which learns its own cognitive and physical behavior is constructed. The physical challenge was minimizing size and weight of the hand which has enough strength and precision to manipulate objects. Commercially available actuators and sensors are chosen, and motor and sensor controllers are designed and constructed. The controller boards are mounted on the dorsum, controlling all the motors and sensors of the hand. When the whole system was integrated, the overall weight of the hand was less than 1.9 pounds. The arm, which is under construction, is capable of exerting about three pound torque at the tip of the hand, without the weight of hand, resulting in one pound maximum load torque.

The cognitive challenge is more complex because the problem itself is not well-stated. With our existing technology and biological facts, very limited implementation was made. Utilizing our knowledge of nervous system organization, low level operation is executed locally at an MC6811 controller, which simulates the spinal cord. It contains a feedback controller which stablizes and minimizes the error of finger positions, and a reflex system for the fingers. The higher level learning schema is designed, trained and tested on MATLAB and later will be implemented on an MC68332 controller for autonomy. It learns to distinguish object hardness using a competitive learning strategy, learns to detect shear using backpropagation algorithm, and learned overall simple grasping using reinforcement learning. All the strategies used are defined with the inspiration of human neural system and human response to given stimuli, but the implementation of them is not necessarily a direct model. This system simulates the surface level learning strategy shown in infants, but may not coincide with human's actual learning process.

5.2 The Future

5.2.1 Physical Work

By building a system, many improvements that can be made are realized.

STRUCTURE: The whole structure of hand can be made even smaller. The pieces are made larger than absolutely necessary for building simplicity. When a part is smaller, the error ratio becomes higher for the same error caused in machining. The diameter of fingers can be cut in half if the pulleys can be machined to fit the need. Motors can be organized as shown in Figure 5-1 so that the size of the palm can be also minimized. For more compliancy, spring loaded joints for proximal joints could be considered to give another degree of freedom of an axis perpendicular to the existing rotation at a proximal joint. The weight can be minimized significantly if the number of screws are reduced by building more complicated parts instead of bolting two simple pieces together using screws.

SENSORS: Tactile sensor technology needs to leap a big step. Sensors need to be aligned at the silicon level, giving a high resolution array of force sensors. The fundamental idea of wires and connectors needs to be improved or changed to adopt a tactile system that can be integrated in a human form.

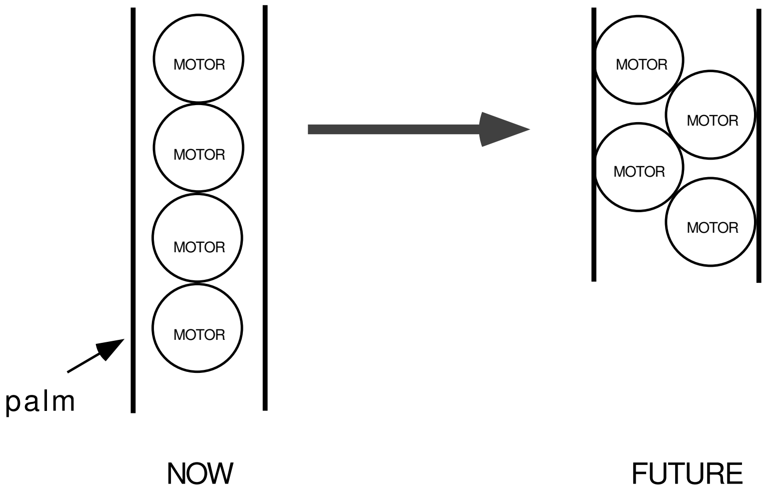

Figure 5-1: A diagram of motor alignment improvement.

COMPUTING: If the motor controller and sensor interface boards design does not require any change, it may be implemented on a chip containing all the capabilities needed. This should significantly reduce the size and weight of the system. Eventually this should be mounted in the spine.

The hand will be soon interfaced with the arm, connecting to the whole body. With arm manipulation capability and the existance of visual and auditory feedback, a door will be opened for building a more complex system that triggers many new constraints and limitations.

Biology has its own amazing system which allows organisms to live and function. It is the duty of scientists to attempt to decode the organic system for a deeper understanding of nature.

5.2.2 Cognitive Work

Neural networks have allowed scientists to take a big step in being adaptive and flexible to the environment which is rapidly changing and is full of noise. However, all the learning theories that are implementable today unfortunately contain many assumptions that may not be true in the real world. For example, they all assume a perfect Markov decision world; the complete set of state information can be accessed by the agent any time.

Human cognition is still a black box that neuroscientists, philosophers, computer scientists and many more are required to keep tackling and investigating. As a tiny step, the attempt to understand the infants manipulation learning by implementating a physical hand was described in this thesis. Babies may not use a learning mechanism close to what is described, but when the whole body is integrated, we may discover a phenomena that could not be obvious before. Studying infants learning system seems to be a suitable starting point since the development of cognition is initiated by the social interactions and learning that occurs during infancy. To get a closer look at cognition itself, a much simpler physical system with minimal cognitive assumptions may need to be build to tackle even lower level cognitive problems.

Afterall, the project to understand human cognition has just started.

Bibliography

[1] Bower, T. G. R., "The Rational Infant: Learning in Infancy", W. H. Freeman and Company, 1989.

[2] Brock, David L. Brock, and Salisbury, J. Kenneth, "Implementation of Behavioral Control on a Robot Hand/Arm System", 1991

[3] Brooks, Rodney A., and Stein, Lynn A., "Building Brains for Bodies", A.I. Memo No. 1439, Cambridge, MA. 1993.

[4] Brooks, Rodney A., "L", IS Robotics, Inc., 1994.

[5] Buekers, Martinus J., Magill, Richard A., and Sneyers, Katrien M., "Resolving a Conflict Between Sensory Feedback and Knowledge of Results, While Learning a Motor Skill", Journal of Motor Behavior, 1994, Vol. 26, No. 1, pp. 27-35.

[6] Chamberlin, Craig J., and Magill, Richard A., "A Note on Schema and Exemplar Approaches to Motor Skill Representation in Memory", Journal of Motor Behavior, 1992, Vol. 24, No. 2, pp.221-224.

[7] Cutkosky, Mark R., and Howe, Robert D., "Dextrous Robot Hands"

[8] Durbin, Richard, Miall, Christopher, and Mitchison, Graeme, "The Computing Neuron", Addison-Wesley Publishing Company, 1989.

[9] Franklin, Gene F, Powell, J. David, and Emami-Naeini, Abbas, "Feedback Control of Dynamic Systems", Addison Wesley Publishing Company, 1991.

[10] Goodwin, A. W., Darian-Smith, I., "Hand Function and the Neocortex", University of Melbourne, Australia. 1985.

[11] Greiner, Helen, "Passive and Active Grasping with a Prehensile Robot End-Effector", S.M. thesis, MIT A.I. Lab, Cambridge, MA. 1990.

[12] Heykin, Simon, "Neural Networks", Macmillan College Publishing Company, New York, NY. 1994.

[13] Hertz, John, Krogh, Anders, Palmer, Richard G., "Introduction to the theory of neural computation", Addison Wesley Publishing Company, 1991.

[14] Jacobson S.C., et Al., "The Utah/MIT Dextrous Hand: Work in Progress", The first International Symposium of Robotics Research, 1984.

[15] Kapogiannis, Eleni, "Design of a Large Scale MIMD Computer", M.Eng. thesis, MIT A.I. Lab, Cambridge, MA. 1994.

[16] Kernodle, Michael W., and Carlton, Les G., "Information Feedback and the Learning of Multiple-Degree-of-Freedom Activities" Journal of Motor Behavior, 1992, Vol. 24, No. 2, pp.187-196.

[17] Kohonen, T., "Self-organized formation of topologically correct feature maps", Biological Cybernetics 43, pp.59-69, 1982.

[18] Lambert, D., et al. "The brain: A user's manual", Berkley Publishing Group, 1982.

[19] Langreth, Robert, "The Bad Boy of Robotics", Popular Science, pp.88 - 91, June, 1995.

[20] Lin, Long-Ji, "Reinforcement Learning for Robots Using Neural Networks", Ph.D thesis, Carnegie Mellon University, Pittsburgh, PA. 1992.

[21] Lundstrom, G., Glemme, B., Rooks, B. W., "Industrial Robots-GRIPPER REVIEW", International Fluidics Services Ltd., 1979.

[22] McCall, Robert B., "Exploratory Manipulation And Play In the Human Infant", Monographs: Society of research in child development, Unversity of Chicago Press, No. 155, Vol. 39, No.2, 1974.

[23] Miller, W. Thomas, Sutton, Richard S., and Werbos, Paul J., "Neural Networks for Control", The MIT Press, Cambridge, MA. 1990.

[24] Napier, John Russell, "Hands", Pantheon Books, New York, 1980.

[25] Narendra, Kumpati S., and Mukhopadhyay, Snehasis, "Intelligent Control Using Neural Networks", American Control Conference, Boston, MA. 1991.

[26] Nolte, John, "The human brain", Mosby Year Book.

[27] Rabelo, Luis C., and Avula, Xavier J. R., "Nierarchical Neurocontroller Architecture for Robotic Manipulation", IEEE International Conference on Robotics and Automation, Sacramento, CA. 1991.

[28] Rader, N., Sstern, J.D., "Visually elicited reaching in neonatates", Chile Development, 53:1004 - 1007. 1982.

[29] Rosenblatt, F., "The Perceptron: A probabilistic model for information storage and organization in the brain", Psychological Review 65, pp.386-408, 1958.

[30] Salisbury, J. Kenneth, and Craig, John J., "Articulated Hands: Force Control and Kinematic Issues", International Journal of Robotics Research, 1982, Vol. 1, No. 1.

[31] Salisbury, J. K., "Kinematic and Force Analysis of Articulated Hands", Ph.D Thesis, Stanford University Mechanical Engineering and Computer Science Dept., Stanford, CA, 1982.

[32] Shirane, Reikichi, "Tsukuba kagakuhaku to Nihon no kagaku gijutsu", Journal of the Robotics Society of Japan, vol. 4, no. 4, p.42. 1985.

[33] Sutton, Richard S., Barto, Andrew G., and Williams, Ronald J., "Reinforcement Learning is Direct Adaptice Optimal Control", American Control Conference, Boston, MA. 1991.

[34] von Hofsten, Claes, "Structuring of Early Reaching Movements: A Longitudinal Study", Journal of Motor Behavior, Vol. 23, No. 4, pp.280–292, 1991.

[35] Watkins, C.J.C.H., "Learning from Delayed Rewards", Ph.D thesis, King's College, Cambridge, 1989.

[36] Wiener, N., "Cybernetics", Cambridge, MA: MIT Press, 1948.

[37] Williams, Ronald J., "One the Use of Backpropagation in Associative Reinforcement Learning", IEEE International Conference on Neural Networks, 1988.