Introducing Liquid Haptics in High Bandwidth Human Computer Interfaces

Tom White B.S. Mathematics
University of Georgia, 1994

Submitted to the program in Media Arts and Sciences,
School of Architecture and Planning,

In partial fulfillment of the requirements for the degree of
Master of Science in Media Arts and Sciences
at the **Massachusetts Institute of Technology**
June 1998

Author

Program in Media Arts and Sciences
May 8, 1998

Certified By

John Maeda
Assistant Professor of Design and Computation
Thesis Supervisor

Accepted By

Stephen A. Benton
Chair, Departmental Committee on Graduate Students
Program in Media Arts and Sciences

Introducing Liquid Haptics in High Bandwidth
Human Computer Interfaces

by Tom White

Submitted to the program in Media Arts and Sciences,
School of Architecture and Planning, on May 8. 1998
in partial fulfillment of the requirements for the degree of
Master of Science in Media Arts and Sciences

Abstract

This thesis presents liquid haptics as a new way of communicating with computational media. Liquid haptics refers to computer interfaces utilizing liquid filled containers and designed to take advantage of the fine tactile sense and motor control of the hands. Here, liquid haptics is implemented using pillow shaped bladders to both measure high bandwidth tactile information from a person and deliver force feedback in the liquid itself. The input system can measure detailed two dimensional information about the hands with a high refresh rate. The output system can provide subtle force feedback cues by dynamically changing the viscosity properties of the liquid substrate. These systems represent a system of interacting with computational media that is both efficient and pleasurable. I describe the design and construction of the liquid haptic systems that were created and show examples of applications where liquid haptics makes sense.

Thesis Supervisor: John Maeda
Title: Assistant Professor of Design and Computation

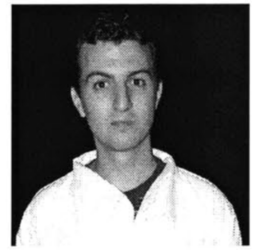
Readers

Introducing Liquid Haptics in HighBandwidth
Human Computer Interfaces

by Tom White

Thesis Supervisor: John Maeda

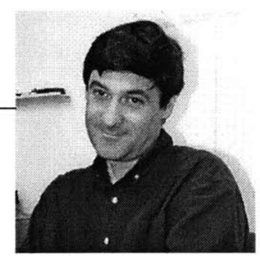
The following people have served as readers for this thesis:



Tom White

John Maeda

Hiroshi Ishii



Joe Paradiso

Hiroshi Ishii
Associate Professor of Media Arts and Sciences
MIT Media Laboratory

Joe Paradiso
Research Scientist, Physics and Media
MIT Media Laboratory

Table of Contents

Dedication

This is dedicated to all the little people that helped me along the way. After all, what kind of man are ye that don't believe in the little people?

1 Introduction

1.1 Motivation

For every task performed on a computer, there is necessarily an interface. The human computer interface defines our experience with the computation. To the layman, the interface is the computer.

But there appears to be no Moore's law of general-purpose human computer interfaces. Computers are called on to assist in production and personal tasks more now than in the past, yet the number of available interfaces in common use remains constant, if not declining from the pressure of standardization. Implicitly, we are accepting a restriction of what is possible in general purpose computation as our means of interaction is assumed in the program itself. For example, a binary two-dimensional selecting device (mouse) is implicitly written into the Java programming language with no means of assuming alternative devices natively available to the programmer. What should ideally be a design decision for the computational creator and personal preference for the operator is here obliterated by the programming language itself.

It is unreasonable to believe that as computers are used for an ever-broadening number of both business and leisure activities, people will continue to be content with one or two possible modes of interaction. Just as a typographer has many different typefaces at his disposal to control legibility and affect, or as people have many pairs of shoes and choose based on mood and utility, an environment immersed in computation should be expected to have a number of possible interfaces with overlapping functions to be used on analysis or whim of the person who is to use it.

Liquid haptics represents a new general-purpose computer interface, and the implementations herein demonstrate their utility. The metric for measuring the success of the interfaces is both in (1) examining if this interface can perform tasks more efficiently or reveal unrealized possibilities in the interaction itself, and (2) the ability of the interface to add value to people's lives by virtue of being more pleasing for different types of tasks, adding variety, or other aesthetic concerns.

1.2 Manifest

- The motivation and implementation of both an input and an output liquid haptic device are presented.
- Several working and hypothetical applications are presented in evaluating the usefulness of these new devices.
- The devices are put into context by examining liquid haptics as it relates to computational art and the other work in the Aesthetics and Computation Group and the MIT Media Lab.
- Promising areas of future work with liquid haptics are discussed.

2 Background

2.1 Purpose

There are countless papers that discuss evaluation and user testing of different new hardware and software, and these provide firm ground for examining the issues in designing a new human computer interface. But care should be taken to remember that no one formula reveals the value of an interface to those that use it. Designing a user interface is about creating a space between a person and a computer, which is a design process similar to architecture. The measured fitness of a device for information transfer between person and computer is not always more important than the aesthetics of the common physical and conceptual boundary created between the computer and the operator.

2.2 Computer Interfaces

One very popular topic in user interface research is Fitts Law, based on information theory for predicting how long it will take a person to point to a target [FIT 54]. The law states that given a target of width W at a distance A, the time T needed to point aimed movements is:

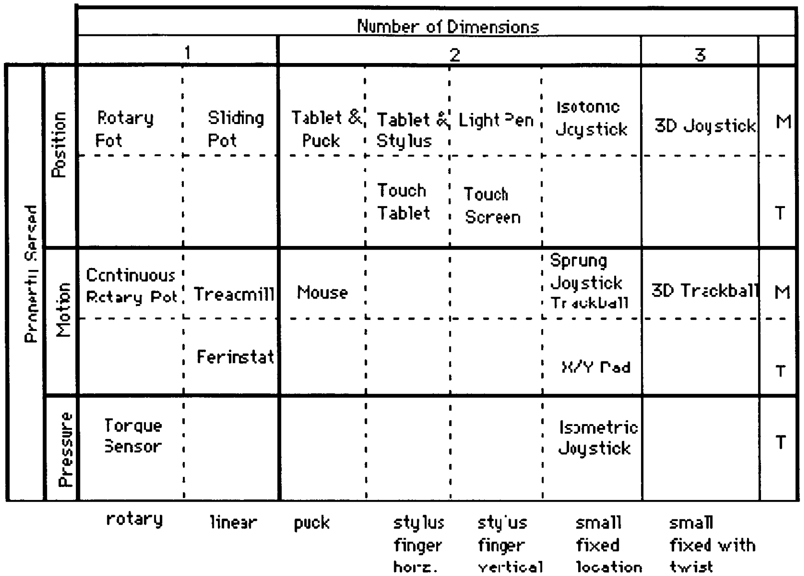
$$T = a + b \log_2 (A / W + c)$$

where a and b are constants.

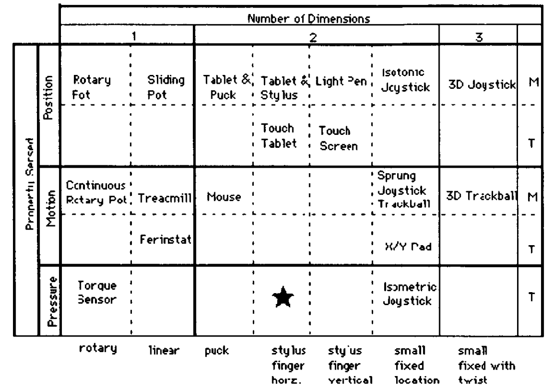
The purported accuracy of this equation coupled with the lure of analyzing such a model in a field hungry for quantitative results probably accounts for its popularity. Perhaps as popular is the methodologically questionable study by Langolf et al which uses Fitts law to get real numbers on the bandwidth of the fingers (38 bits/second), wrist (23 bits/second) and arm (10 bits/second) [LAN 76].

While Fitts law is useful in that it allows one to predict the accuracy versus speed trade-off in motor control, of more long term value is the broad connection he established between interface design and information theory. Picturing humans and computers as information sources offers a new perspective on designing interfaces between them, and the mathematical models of information theory offer a framework for measuring the communication efficiency of the system as bandwidth.

Other analyses in human computer interfaces include dissecting the possible interactions into several parameters and creating a classification system based on this framework. A good classification system is useful in both the analysis of existing systems by supporting valid comparisons, and in exploring new types of devices that could be used for in computer interaction. An important taxonomy was introduces in 1982 by Bill Buxton, notable here because it separated out the pragmatic level, which encompasses issues of gestures, spaces, and devices [BUX 82]. In his own words, "Our main thesis is that since the primary level of contact with an interactive system is at the level of pragmatics, this level has one of the strongest effects on the user's perception of the system. Consequently, the models which we adopt in order to specify, design, implement, compare, and evaluate interactive systems must be sufficiently rich to capture and communicate the system's properties at this level." Buxton shows how choice of devices implicitly effects the usability of an interface, and how device independence is programmer's abstraction of limited use. He builds on this idea to develop a taxonomy which he neatly lays out in a tableau of existing and potential input devices, and discusses how the table can be used to make analogies and predict future input devices in a manner similar to Mendeleev's periodic table of elements. This classification system has more recently been extended to a 10 dimensional hypercube of device characteristics [LIP 93].



Buxton's Taxonomy of Input Devices categorizes continuous manual input devices. First order classification is by property sensed (rows) and number of Dimensions (columns). Second order classification of rows separates encumbering or mechanical ("M") devices from those that are touch sensitive ("T"). Second order classification of columns group by comparable motor control in operation.



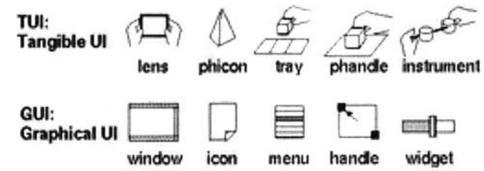
Interestingly, liquid haptic input devices fill a then blank location on Buxton's tableau. The star indicates how liquid haptics best fits into his taxonomy.

Elsewhere in the Media Lab, much work centers around computer interfaces. For example, the Physics and Media group states it's purpose as studying "many of the most serious challenges and significant opportunities in information technologies [that] lie right at this interface" [PHY 98]. Other notable work within the lab focusing on computer human interfaces include constructing interfaces that couple physical objects to digital information, [ISH 97], developing wireless, full-bodied interactions with autonomous agents [MAE 96], creating adaptive user interfaces based on speech and natural language understanding [SCH 94], and creating interactive three-dimensional holographic displays [LUC 95].

Lost in much of the discussion of computer interfaces is a central issue: do people actually enjoy using the interface? In the introduction to his book, Myron Kruger states the purpose of his work as a problem in aesthetics: "What are the various ways in which people and machines might interact, and which of these are most pleasing?" [KRU 90] Though a paper introducing an input device rarely spends time discussing the pleasure that people have using it, one hopes the designer has seriously considered this issue since the device is created ultimately to serve people. The ultimate goal is to create a computational process that is both useful and gratifying, and the most precise instrument designed for human computer interaction is a failure if it is chronically unsatisfying. Krueger doubts that a technologist who disdains aesthetics can design a humane technology.

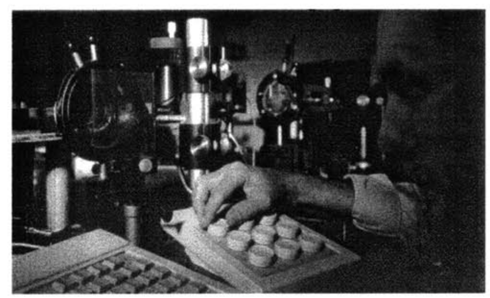
2.3 Haptics

In a summary paper on haptics interaction projects, Brooks defines haptic as "pertaining to sensations such as touch, temperature, pressure, etc. mediated by skin, muscle, tendon, or joint." [BRO 90] Buxton says that haptic input is input that involves physical contact between the computer and the person, which can be via



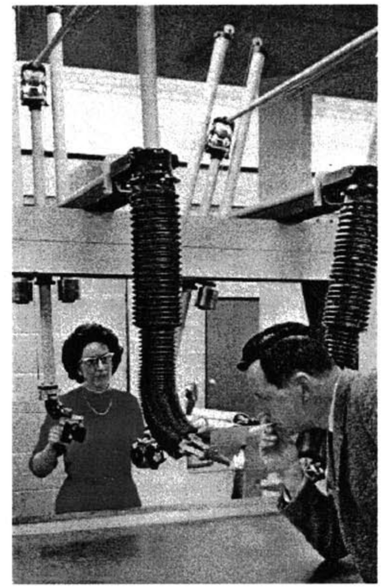
Ishii's work includes building physical instantiations of traditional graphic user interface elements.

Maes and others developed a vision-based system for interacting with autonomous agents.



The Spatial Imaging group at the Media Lab has developed a holographic video display capable of interactively rendering full-color images with horizontal parallax.

the hands using a mouse, feet using a pedal, or even the tongue using a special joystick. Further, Buxton asserts, every haptic input device can also be considered to provide output through the tactile or kinesthetic feedback that it provides to the user [BUX 95].



Early teleoperator system with mechanical kinesthetic force feedback.

Haptic force feedback interfaces provide output explicitly. These generally deal with combining tactile and kinesthetic perception to create a feedback channel, usually based on perceiving real world objects. Kinesthetic perception senses movement or force in muscles or joints, and the tactile perception senses shapes and textures.

One area of haptic research is haptic displays. Kinesthetic displays generally use robotic feedback to manipulate objects in remote or virtual environments, and were first developed in the field of teleoperation [HAN 95]. Tactile displays recreate skin sensations to assist manipulation and exploration tasks, often using high-frequency vibrations, small-scale shape or pressure distribution, and thermal properties [HOW 97].

Within the Media Lab there is no one area that concentrates on haptic research, though individuals in various groups incorporate haptic displays and force feedback into their research work. Past interdisciplinary Media Lab work has investigated simulating the rich and complex world of surface texture and material properties by creating a system that can synthesize texture via a tactile force feedback display [MIN 95]. Other work in the Tangible Media Group has concentrated on creating novel haptic displays as a means of conveying touch and presence over distance [BRA 97]. Elsewhere in the lab work has been done developing tactual displays to convey directional lines and simple geometric patterns for use in wearable computing [TAN 97]. And separate from these is very recent work being done in the Spatial Imaging Group to create real time virtual lathe by coupling a holographic video with a handheld kinesthetic force feedback display to create a coincident visuo-haptic workspace [PLE 98].

Brave's inTouch roller system serves as a tangible telephone, enabling haptic interpersonal communication across distance.

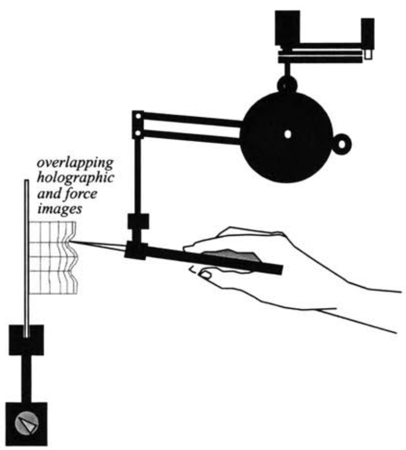
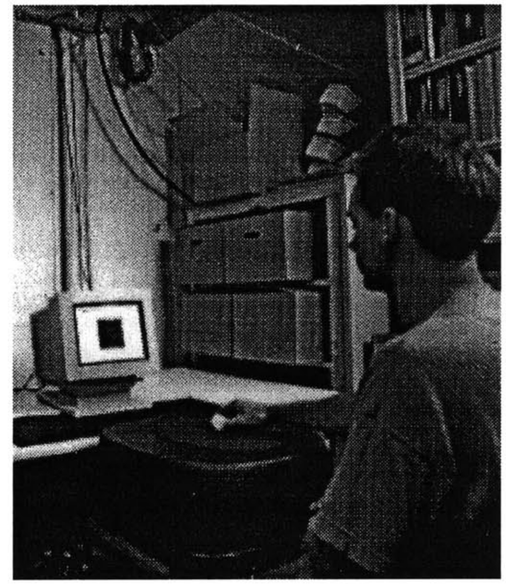
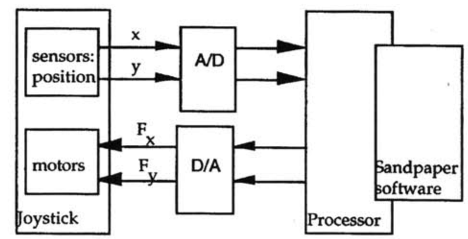


Diagram of the coincident display in [PLE 98] in which a person carves on a virtual lathe using a stylus by interacting with the holographic image while feeling forces that result from a force model.

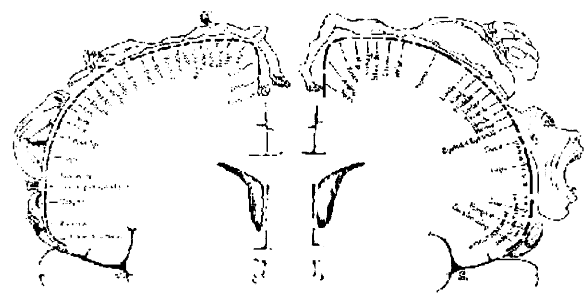




Minsky's Sandpaper system in use and a block diagram of the system.

2.3 Human Anatomy and the Hand

The most logical parts of the body to engage with a computer are those with which we have the most dexterity, sensation, and control. Neurophysiological studies have shown that different areas of the human body have disproportionate anatomical brain area that would be expected if based on physical size and mass. This is true in both the somatosensory (input) and motor cortex (output) areas of the brain, and provides a basis for targeting various areas of the body based on expected input and output performance in an interface. A homunculus model of the brain shows the fingers and hands dominate the motor cortex and are also well represented in the somatosensory cortex.



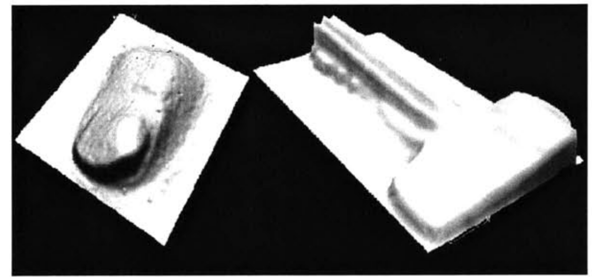
Homunculus model of somatosensory (left) and motor (right) cortex [PEN 50]

Many studies have evaluated the information processing ability of different parts of the body, but they usually center on the task of pointing and binary selection. Though the hands generally, but not always [RAV 97], score well at these tasks, the hands have much more potential as a high bandwidth interface when subtle differences in placement and pressure can be recorded.

Finally, man's intelligence as species is probably not independent of his dexterity and ability to examine the world around him. An interesting hypothesis in cognitive science suggests that one of the driving forces behind the evolution of human intelligence was man's ability to do fine manipulation of objects with his hands [PIN 97].

2.4 Concurrent Work: the Haptic Lens

During this development of liquid haptics, a device appeared that is quite similar to the implementation herein of an input device: the haptic lens [SIN 97]. Developed at the Interactive Media Technology Center at Georgia Tech, the haptic lens certainly fits within the taxonomy of liquid haptics. The haptic lens enables the real-time visualization of the haptic sense of pressure, and applications developed for the haptic lens include palpation, robotics, 3D digitizing, and a novel 3D input device.



Surface mapping of a finger and key using the Haptic Lens

The design of the haptic lens is very similar to the implementation of the liquid haptic input device presented in the following chapter. Several design constraints were shared between devices, though solutions vary in consistency. Both pads use a material with an optic gradient lined with white material opposite a camera and light source. Both pads pay special attention to reducing reflections from on-axis illumination, though the optic lens uses cross-polarized filters as a solution and mine used a careful arrangement of light sources. One more important difference is instead of containing an optically dense liquid, the haptic lens uses a compressible optically attenuating grey dye for a more rugged and precise device. The haptic lens is quite remarkable in its fidelity - some smaller pads can register individual threads in a 1/4" screw. This reflects the primary design goal of accurate reconstruction of three-dimensional surface information under a specific pressure.

2.5 Personal Interface Work

Liquid haptics is not the author's first experience in programming and designing new computer human interfaces. I have been working with novel computer interfaces for several years before developing the ideas behind liquid haptic devices. Comparing these past interfaces with current work will explain the development and design decisions of my current work.

Scene from *Small Planet,* Kruger's exhibit in the 1993 SIGGRAPH Art Show. Here the author is flying over a mountain range flying away from the planet. The body is being tracked in real time, and the arm gestures control direction and lift of the view on the projector.

My first large interface project was as a collaborator in Myron Kruger's Small Planet exhibit in the SIGGRAPH 1993 art show. In this exhibit, people could fly over a planet by making simple flying gestures with the arms and the first person perspective of view would be projected in front of them. As an intern, part of my job was to write the code that interpreted hand position from Kruger's VIDEOPLACE hardware as velocity changes over a modeled terrain. I was drawn to Krueger's work in part because I share his desire to create interfaces that are unencumbering, not requiring a person to wear or hold onto any devices or sensing technology. Unencumered devices are conceptually lightweight in that that they can be used both active and passive activities and they don't require any overhead in engaging, disengaging, or changing the level of attention given to an application. Over the next few years I did other computer installations, including my own exhibit at SIGGRAPH in 1995, and these usually centered around using cameras to track the body.

After arriving to the Media Lab in 1996, I did some work with the fish sensors developed in the Physics and Media group. The fish sensors use capacitive sensing to estimate limb position and do not require any physical contact with a person's body [SMI 98]. Though the technology was very different, from the point of view of the person using the interface the FishPad recognized body gestures without requiring any tethering devices similar to my previous work with camera based interfaces.

Injection, the author's 1995 SIGGRAPH exhibit, also used cameras to track people at a distance over a large floor area.

An interesting point about both my vision and FishPad interfaces is that they were completely non-haptic in that they did not require any contact whatsoever with the person using them. This puts these devices in a small category of input devices that operate devoid of human touch, alongside voice recognition and eye tracking instead of with traditional desktop interfaces like a mouse, keyboard, or stylus.

In my FishPad research, the application domain was very different from my earlier camera based devices. I was not designing a short interactive art installation, but instead trying to develop a device that could be used as an input device for common tasks. Often these required more accuracy and use for longer periods of time. I found that controlling computational tasks with arm gestures was relatively inaccurate and fatiguing over time.

Here the author works with an early version of the two handed FishPad. The metal bracelet was eliminated in later designs .

After working on alternative designs for the FishPad I began to rethink the problem of designing an computer interface. I wanted to maintain the idea of an unencumbered, lightweight interface. But instead of starting with the technology I began to think about designing a device around what people found natural and relaxing. It was this work that led to the first liquid haptic devices.

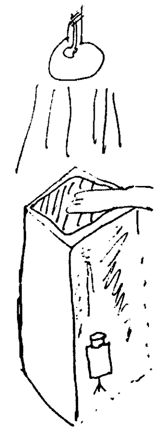
3 Liquid Haptic Devices

My aim is to create computer interfaces that are both efficient and satisfying. On the human body, the most logical target is to concentrate on the hands. Ideally, the computer would have instant and continuous readings of the location of the hand as well as be able to provide force feedback output to the hand as it moved. Decomposing this problem, a first liquid haptic device was created as an input device for the hand. This was followed by a separate hand-based output device.

3.1 Initial Liquid Haptic Input Pad

In order to enable several different types of gestures and potential hand positions, especially the natural grasping curl of the fingers, I looked for ways to capture complete pressure information about hand position at high refresh rates. With a transfer rate of roughly one billion bits per second, a camera is one of the highest bandwidth input devices common on personal computers. Thus, an optical liquid haptic input device could be prototyped. A special camera-based system was built in which the camera aimed upwards at a clear bag of liquid, and several liquids were tested to see how they interacted with the camera and felt to a person using hand pressure. Surprisingly, soy sauce was found to have several desirable properties, including opacity with gradations at different thicknesses, evenly distributed color that did not settle, very low viscosity so that it felt springy and responsive, and the liquid itself is both readily available and non-toxic.

In this initial implementation, a 6-inch by 4-inch bladder was constructed from clear plastic bags. The camera looked up at the bladder with a light overhead.



Sketch of initial prototype where light from above is transmitted through the liquid to the camera below in the box when the pad is compressed.

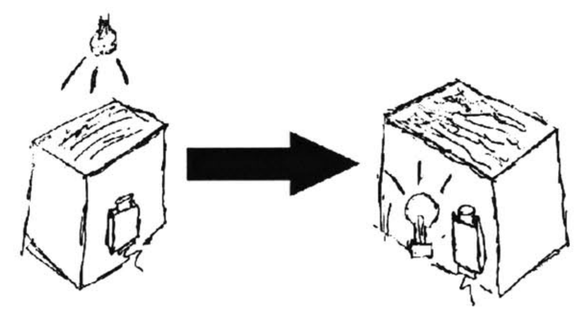
When the person pressed down on the liquid, the light would shine through the bag and pressure information could be inferred.

Though successful in proving the overall concept, this configuration was problematic. The optical information in this configuration created problems, mainly because the areas between fingers would flatten out, letting light through while the fingers themselves were opaque. Opaque fingers created sharp edges of blocked light, including obscuring the highest pressure at the point of contact with the fingertips. The bladder itself threatened to breach because of the weakness of over-the-counter plastic, which necessitated layering. The poor quality of this plastic combined with the layering of material reduced the tactile experience of interacting with the liquid by calling attention to itself.

Raw data as captured from the first input pad.

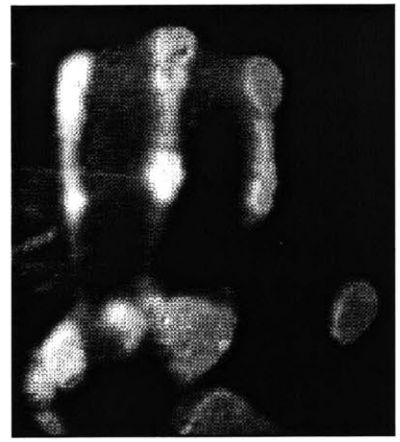
3.2 Revised Liquid Haptic Input Device

The revision of this interface again used soy sauce, though this time I used a single heat-sealed medical bladder inset into a larger box. The optics were reversed so that the camera used light reflected from a white surface layered above the liquid container instead of simply passing through it. The construction of this device was more challenging because of several optical surprises, including false readings from internal reflection. But this approach was superior in highlighting the anatomy of the hand as it moved and the relative pressure of the liquid. The tougher bladder ran little risk of puncturing. When overlaid with a flexible material, it felt softer and more coherent than the earlier design.



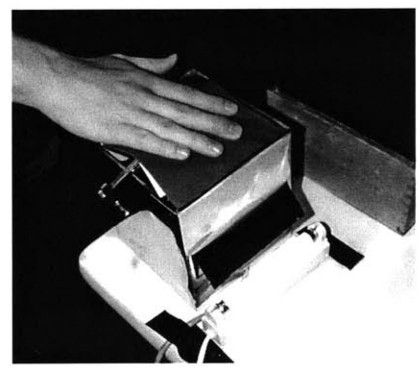
Sketch of design changes in the input pad.

Yet this design still had several areas that could still be improved. The optical problems with internal reflection caused the design to have uneven lighting and an overall loss of dynamic range. Though the bladder was covered with a softer membrane, the rigidity of the plastic caused the overall tactile effect to be somewhat stiff. And, as with most plastics, the bladder was semi-permeable and over time allowed the scent of soy sauce to escape, which was distracting at best.

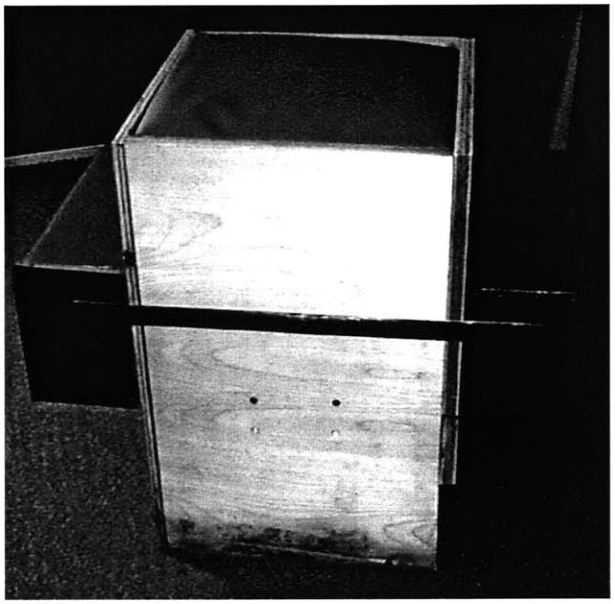

Improved raw data from reflection based optics.

3.3 Final Liquid Haptic Input Device

The final design redistributed the lighting more evenly to improve the dynamic range of the device. More radically, a variety of new usable bladder materials were used, including silicone, latex, urethane, and polyurethane. Many of the new bladder materials are poured from mixtures into molds, which allows fine control over the thickness and size. The silicone used for the top surface of the bladder where the hand is placed is tough yet much more soft and less rigid than the plastic. These materials also allow me to continue using the easily available soy sauce despite its odor, as they are not as permeable as the plastic.

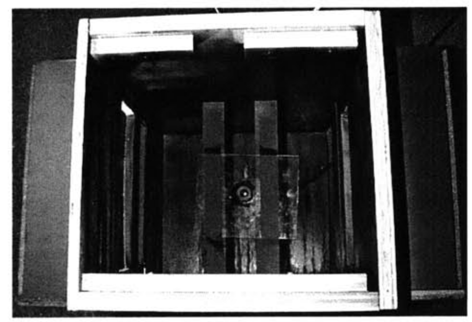

A scaled down version of the liquid haptic input pad for travelling.

Finally, the input liquid haptic input pad was scaled down into a "travel version" of the original. Though originally constructed to facilitate travel to a talk, the small liquid haptic pad introduced some interesting issues in device scaling. Smaller devices would generally have a thinner bladder and require less liquid, but it is desirable to have complete opacity when the bladder is at rest. For this reason, the optic density is dependent on the overall size of the input pad. As soy sauce loses fidelity with thinner bladders, thicker liquids like inks are preferable.



Final prototype for the input pad as viewed from the front.

The final version of the output pad used internal reflection and side lighting to eliminate reflection of light into the camera.



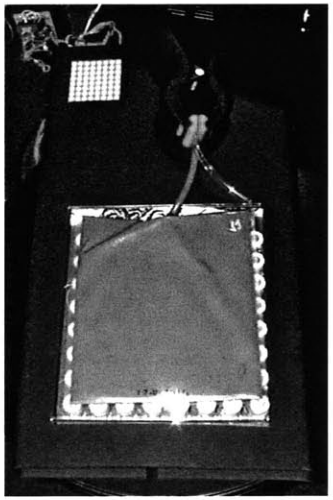
Inside of final input pad. The long lights along the left and right light the underside of the bladder (here removed) evenly, but are placed high enough so that their reflection cannot be seen by the camera.

3.4 Liquid Haptic Output Device

Magneto-rheological (MR) fluids are suspensions of micron-sized magnetizable particles in an oil. When a magnetic field is applied to the liquid the consistency changes proportionally and almost instantly, going from roughly that of motor oil to that of peanut butter. MR Fluids actually develop a yield strength and behave as Bingham solids when a magnetic field is applied. This change can appear as a very large change in effective viscosity and occurs in less than a few milliseconds [LOR 97].

The MR fluid has properties that suggest that it would able to convey distributed force-feedback sensations back to the hand. The liquid would not be able to apply an upward force to the hand in the manner of a kinesthetic display, but it could dynamically change its resilience to applied pressure. This would categorize this device as a tactile display, though traditionally these devices are small and with resolution spacing down to a few millimeters to match human perception abilities.

A bladder of water-based, non-toxic MR fluid was mounted above a grid of 64 electromagnets that could be computer controlled. Details on this MR fluid can be found in Appendix A. This grid of electromagnets are spaced at one-inch intervals and arranged in a checkerboard pattern to maximize the field above the surface. This causes the liquid above to locally change viscosity. The magnetic field can remain constant to create areas of hard and softness, or cycle at different frequencies to create a vibrating texture or exaggerated dynamic feedback. It is worth noting that this design specifies a force-feedback device with no moving parts.



The final liquid haptic output device.

The desired result differs from much previous haptic display work in that I am not trying to simulate the tactile experience of the hand. The idea is to hint at the structure of what is being felt by creating a hill, hole, or rough spot in the space. The purpose of this haptic liquid is to give the programmer some ability to explicitly control feedback mechanism and hint at the structure of what is being manipulated. It was decided to initially test the feasibility of this pad as a force-feedback device without integration with the input pad, in part because the optical properties of the MR fluid are not usable with existing designs.

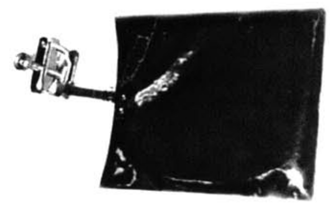
Output pad with bladder removed reveals a grid of 64 electromagnets and electronics to drive them from a desktop computer.

3.5 Bladder Construction

Over time, more and more resources were spent on the design of the bladder itself. Ultimately, the materials were made in house to get the best tactile experience and satisfy the desire to make small modifications to the bladder shape and thickness. But the construction of an optically and tactually satisfactory bladder from plastics is challenging.

The simplest bladders can be made with sandwich bags. Bags such as Ziplock Freezer bags can be layered to create a quick and cheap usable bladder. But the bladder should either be used for transmitted light through the liquid, or the top surface layered with a reflective material. The tactile qualities of these plastics are poor, and the bags can rip with rough use. Nevertheless, this is useful when testing new liquids or prototyping new configurations.

Medical bladders were very successful. Strong bladders can be ordered filled or with a filling tube mounted at one end. The bladders can be ordered with different materials above and below, specifically, a clear bottom and white top as is needed by the optics. But the quality of design was not consistent, and the prefilled bladders usually contained dimples and wrinkles that were disagreeable with the reflection optics. The bladders were generally overlaid with a soft, opaque material to eliminate light transmission and provide a softer tactile contact.



Medical bladders were durable and useful for prototyping different liquid haptic devices.

A better bladder can be constructed with plastic sealing epoxy, glass, and a sheet of latex. The latex is sealed directly to the glass using the epoxy, and the liquid is sealed between the two. The latex is durable, thin, and quite flexible. But it is not soft and is very translucent, which creates problems with transmitted light. A solid seal between the glass and latex is essential to prevent leaks, so testing and reinforcing seals is essential.

A bladder with excellent tactile and optic properties can be made from a thick sheet of glass and a soft but tough white silicone (V1068, Appendix A). The silicone does not bond well to glass, but can be reinforced with epoxy or other seal-ants. Petroleum jelly is put on the glass to create a bulge, then the silicone is poured over the material except for a small opening used later for filling the bladder with liquid. After drying and demolding, the jelly is flushed through the small opening, filled with the final material, and the small opening sealed. The entire seal can be secured with more silicone sealant after the bladder has been filled.

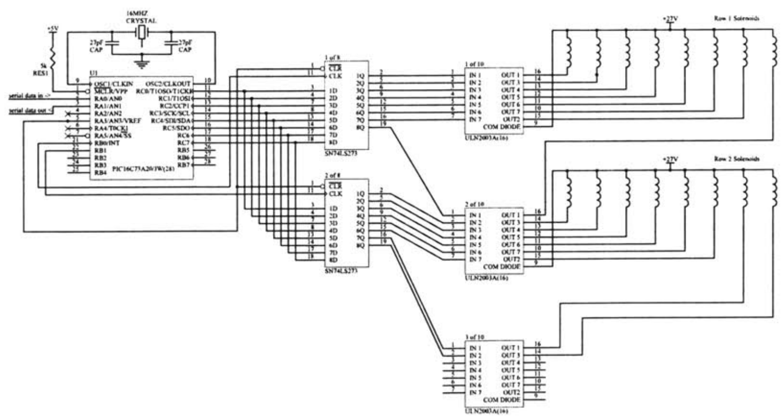
3.6 Magnetic Grid Construction

The liquid haptic output device runs on custom hardware designed by my comrade Paul Yarin [YAR 98]. This hardware connects to a computer via a serial connection. The computer issues commands to the interface that either set the state of a particular cell or the entire pad. The possible states for each cell are on, off, or a frequency with which to vibrate at an even duty cycle. These commands are delivered to a 16C73 PIC chip, which maintains the state of all the magnets.

The Liquid Haptic output pad is driven by a planar grid of 64 coaxial electromagnets. The magnets are mounted on an 8x8 grid with 1-inch center-to-center spacing. 32 of the electromagnets are wired with a north-oriented polarity; the remainder has a south-orientation. These two types are interleaved in a "checkerboard" pattern, such that the four magnets directly adjacent to any given one have the opposite orientation. This way, field couplings between pairs of magnets can be maximized.

Magnetic grid under output pad

The electromagnets are driven by a set of transistors on a simple circuit board. These provide the electromagnets with high-current power, but can be controlled by low-power signals from the electronic circuitry. As each electromagnet must be addressed independently, a separate driver transistor is used for each of the solenoids. ULN2003 Darlington transistor arrays were chosen for their compact packaging, protection from inductive loads, and support for loads up to 350mA.



Paul Yarin's wiring diagram of the first two rows of the output hardware. A larger version of this diagram can be found in Appendix A.

The transistors are controlled by a set of eight 74LS273 octal latches. Each of these latches can "sample and hold" data from the PIC microcontroller. In this way, the 64 separate transistors can be controlled using only 16 lines of data from the PIC. Each latch is used to drive a different row of the display. The data inputs of each latch are wired in parallel, such that the same eight bits of PIC data is sent to each latch. Each latch also has a separate "enable" line, also connected to the PIC. When this line is low, the output of each line mirrors its input. When the line switches high, these output levels are retained, even if the inputs change.

The output levels are driven by a separate 24 volt, 30 amp power supply capable of driving all 64 magnets simultaneously. Thus, the PIC has separate binary control over activating each of the electromagnets in the grid and can vibrate individual cells at up to approximately 150 Hertz while attending to the external serial line.

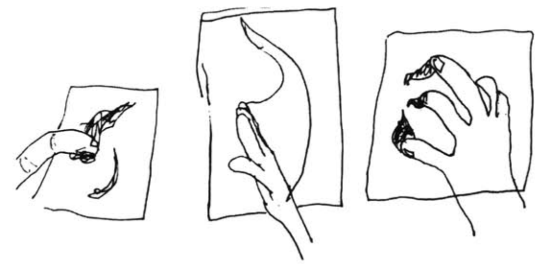
4 Liquid Haptic Applications

This chapter describes the liquid haptic interfaces in action by describing some of the applications that were developed and run with liquid haptic devices. The non-discrete nature of a high refresh, high-resolution device presents different restrictions on the computer programmer, potentially opening up a new domain of software interfaces.

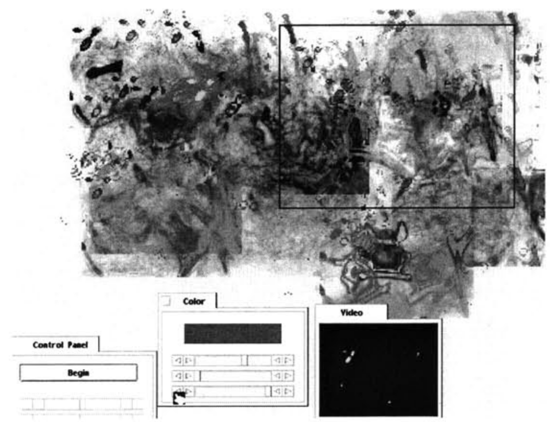
4.1 Fingerpaint

The first program was a fingerpaint application and used the liquid haptic input pad. A person selected a color with the mouse, and then painted that color with the pad. The input area mapped directly onto the painting canvas, and the pressure applied by the hand was proportional to the amount of paint applied to the canvas in that location. The program was an attempt to capture the analog, multi-point, and continuous nature of finger painting.

Fingerpaint was a very straightforward mapping of the liquid haptic input pad to computer graphics as pressure was converted into color on a two dimensional canvas. Several dozen people painted with the program, and most found it fun, natural, and surprisingly relaxing. The process of painting was a good match for the liquid haptic pad because of its non-linear, analog, and expressive nature. Even quick and simple paintings had an appearance that was more complex and continuous than could be expected of usual interfaces. The main complaint was that it was hard to see where the pressure was being applied until the paint was actually applied to the canvas.



It is rare for someone to finger paint with just one finger. More common is using larger areas like the broad side of the hand or multiple fingers simultaneously, and to use varying pressures to get different effects.



Screenshot from the first fingerpaint program.

After entertaining several ideas for modification to allow people to see the hand pressure directly, a solid solution was found. Fingerpaint was extended from two dimensions to three, with the canvas bulging upwards in the location of pressure. Viewed perpendicular to the operator, the program appeared the same, but the canvas could be rotated in three dimensions so that the immediate pressure information could be seen. The result was excellent: it gave more information to the person painting with the liquid pad and changed the feel of the interaction to a dynamic colorful sculpture.

A snapshot from the three-dimensional paint program with a hand pushing on an input bladder superimposed over the painting.

4.2 Stream of Consciousness

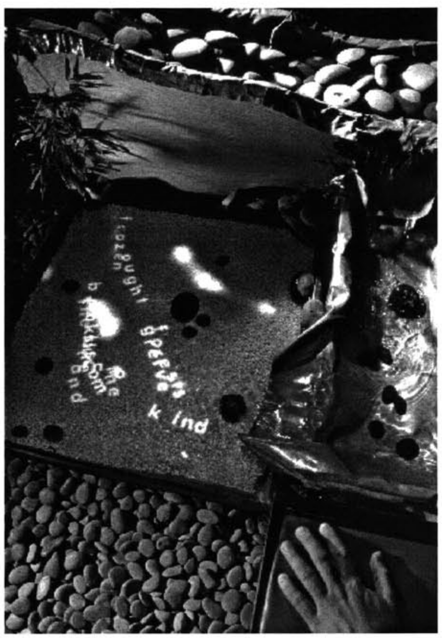
The Stream of Consciousness project was a poetic interactive garden I worked on in collaboration with David Small. The project attempted to bring the computer into the garden in harmony with stone, water, and plant materials. A computer is used to drive a video projector, creating the illusion of text floating on the surface of the water as it flows through the garden. Through the liquid haptic input pad, the person could reach into the pool of words and interact as they flowed by.

The Stream of Consciousness project was designed to be relaxing. To this end, I decided to keep the interaction of the person with the words through the liquid haptic interface straightforward and non-modal. This proved challenging as people naturally want to grab, pull, and dam up the words that are flowing by, and they want to know exactly where their hand on the liquid haptic input pad maps onto the word flow at all times.

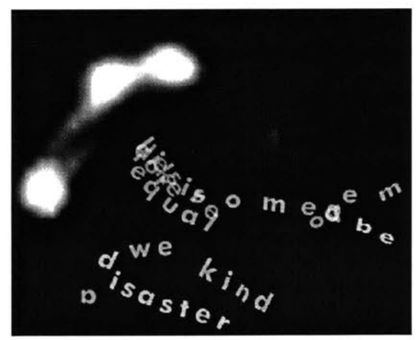
A layout sketch of the Stream of Consciousness showing the stream, overhead projector, and input pad.

To give the person an indication where his hand mapped onto the physical water, the optic input was projected behind the words in a soft blue light and the refresh rate was kept sufficiently high to couple the hand movements with the projected light. The words appear to be repelled by the light, making it easy to create blockages in the flow. This is achieved by having the words slide down the pressure gradient from high to low areas of pressure. If words are selected by grabbing them with the liquid pad, they grow and split into a series of related words. This coexists with the gradient flow action since a word selected near its middle will have roughly equal gradient falls in all directions, and the speed with which a word grows is proportional to the pressure applied at that area.

Using the broad side of the hand is a natural gesture to dam up flowing water. Pressing on an elongated graphic like a word is easily done with multiple fingers.



A stream of words flows through the stream of water; the hand on the liquid pad interface in the foreground is influencing the words.

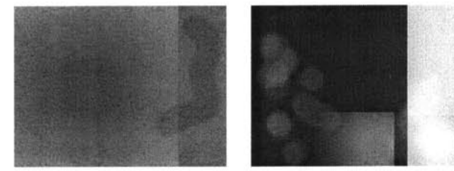


Three fingers hold back a stream of words.

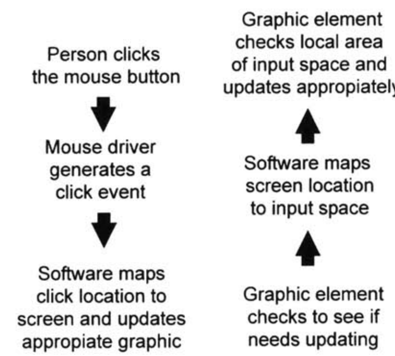
Creating the illusion of words flowing through our installation and that appeared to respond to the input in a natural way was challenging. Dave Small and I created a realistic model of word flow using masses and springs that gave the illusion of words flowing in water. But we still had the problem of making the words appear to be flowing through our particular fountain. To allow easy prototyping, we created software that read in graphics files and interpreted them as vector, dampening, and turbulence fields. In this way we were free to change the physical water flow, and then use a graphics program to quickly paint in the new physics. But this still left the problem of how to incorporate the input pad.

In contrast to traditional event based programming in which each device issues events whenever it changes state, the liquid haptic pad is a high bandwidth device capable of changing state continuously, potentially on the order of megabytes of change per second. For this application, it was not important to analyze the hand information to generate forces unless it potentially interacted with words. So in stead of taking the device centered approach, we had each word map itself into the input space of the pad and examine only the information local to that location. Thus, if a hand falls on the input pad, and there is not a word there to feel it, it has not really made a force. This was far more efficient than doing vision analysis of the entire input space for the number of words we used. This approach represents a useful shift of perspective in programming practice for high bandwidth devices.

The overall effect of the garden was foremost as a quiet, contemplative space. Over time, several hundred people experienced this interaction, and the response was warm and enthusiastic. We were also pleased to see the many unique expe-



Two images used as physics data in modelling the Stream of Consciousness. On the left is a dampening field; lighter shades represent areas of higher friction. At right is a turbulence map in which lighter areas are subject to stronger random force vectors.



Most programs use an event driven model of handling devices in which changing the state of the device propigates to its effect. Because we are using a high bandwidth device, it is more efficient to reverse this model and map the graphics elements into the input space of the pad.

riences because the interaction through the liquid haptic device is so open-ended. Some people were content to passively watch the words, others would repeatedly damn up the words into clumps and release them, and other people would attack the words so that they divided out of control and littered the water with hundreds of words. Even pre-verbal children were able to explore the water and stones through the liquid pad to create "lights" which shone on the water.

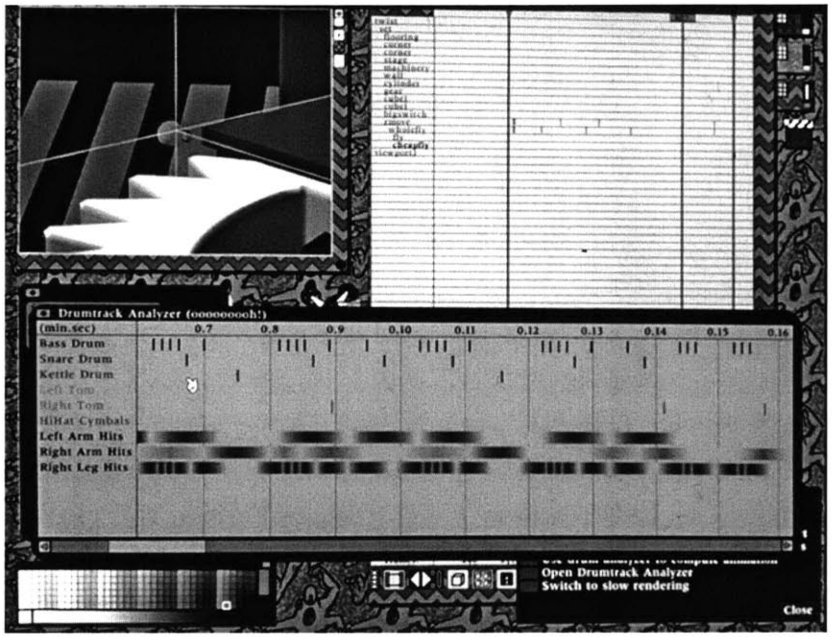
The liquid haptics input pad being used in the Stream of Consciousness installation.

4.3 WorseWindows

Writing programs for the liquid haptic interfaces is tedious because each program is a separate monolithic entity in which all of the routines to handle the interfaces must be written from the ground up. What is needed is a virtual environment which makes fewer assumptions about the existing hardware, is graphically rich and inherently three dimensional, able to accept new code to treat the interfaces as any other input or output device for modularity, and built for design sketches with quick prototyping. Since such a system did not exist, Dave Small and I have begun building one.

A gesture like sliding a sheet of paper along a table involves a distinct sweeping motion with the hand. A similar gesture could be used to slide space, rotating a graphic cylinder around you.

The system is currently called WorseWindows in tribute to a similar effort several years ago by past student Bob Sabiston and his work on BadWindows. BadWindows was a complete windowing environment that was being developed in the Visible Language Workshop in the Media Lab at roughly the same time that X-Windows was being developed elsewhere on campus. At the time BadWindows was groundbreaking, including such luxuries as anti-aliased font libraries.



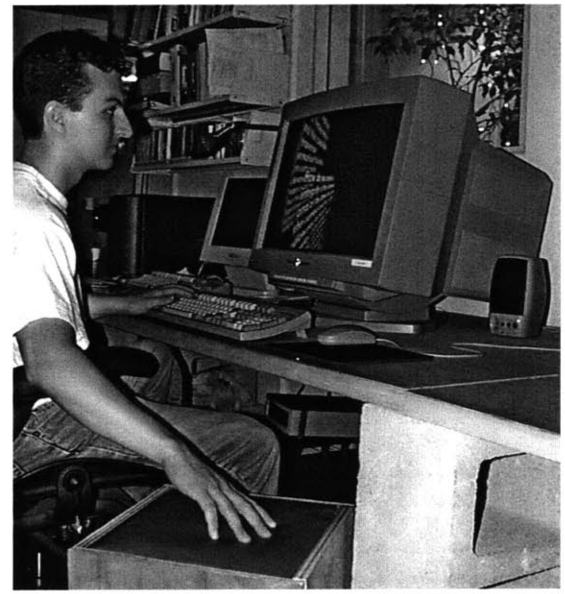
Screenshot from a BadWindows session including multiple applications and antialiased fonts, taken June 1988.

Bob Sabiston at work in his BadWindows environment, 1988.

WorseWindows is intended to be a similar endeavor in a virtual three-dimensional environment. Mock-ups have been created which allow the liquid haptic input pad to be the primary input device in this graphical user interface. In these, the environment is projected in front and two the sides of the viewer in an immersive environment. This allows the peripheral view to be grossly enlarged so that a person has full 360-degree awareness of his environment, similar to the ocular effect of being a bird. The entire cylinder of awareness can be rotated by a simple hand gesture to move items from the periphery to the foreground.

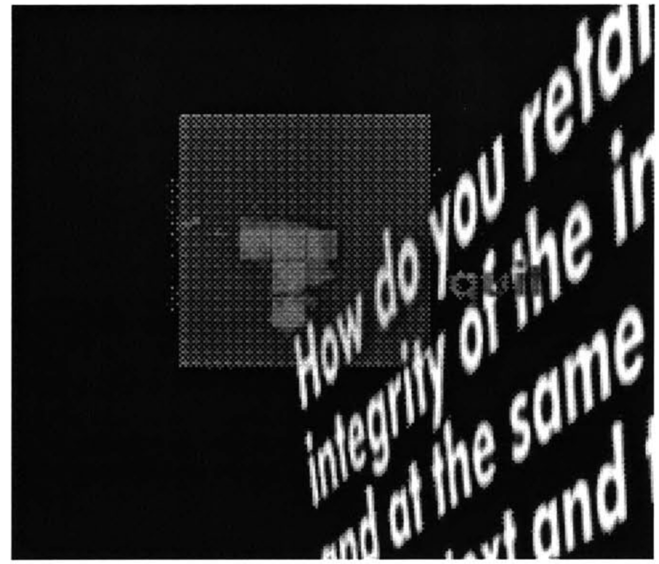
This work shows some new possibilities of the interface. It also is the first application to suggest gesture recognition, albeit crude, which allows a person to perform a traditionally modal task without breaking out of the current task. Richer vision algorithms would allow for several such gestures to be recognized relieving the overhead of switching from mode to mode in operations.

A text rich WorseWindows application currently in development by David Small.



The author recently at work with WorseWindows and a liquid haptics input pad.

Two paragraphs are rotated into focus in the WorseWindows environment by hand movement on the liquid haptic input pad. The pad displays its state via a small transparent overlay which animates as the pad is used.



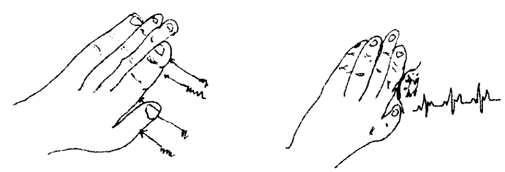
Detail of WorseWindows environment showing the transparent animating overlay icon representing the liquid haptic input pad.

4.4 Wild Leprechaun

Wild leprechaun is a playful interaction with the output pad in which the person tracks a small virtual leprechaun via force feedback haptics. The leprechaun moves around under the hand of the person, breaking up and reforming, vibrating and darting around the space. When the leprechaun stays still, the person can feel his heart beating, and before he runs off again the person can feel the heart accelerate, anticipating the motion. Simultaneous to the haptic interaction is a moving graphic that uses color and movement to simulate the same movement of the virtual leprechaun.

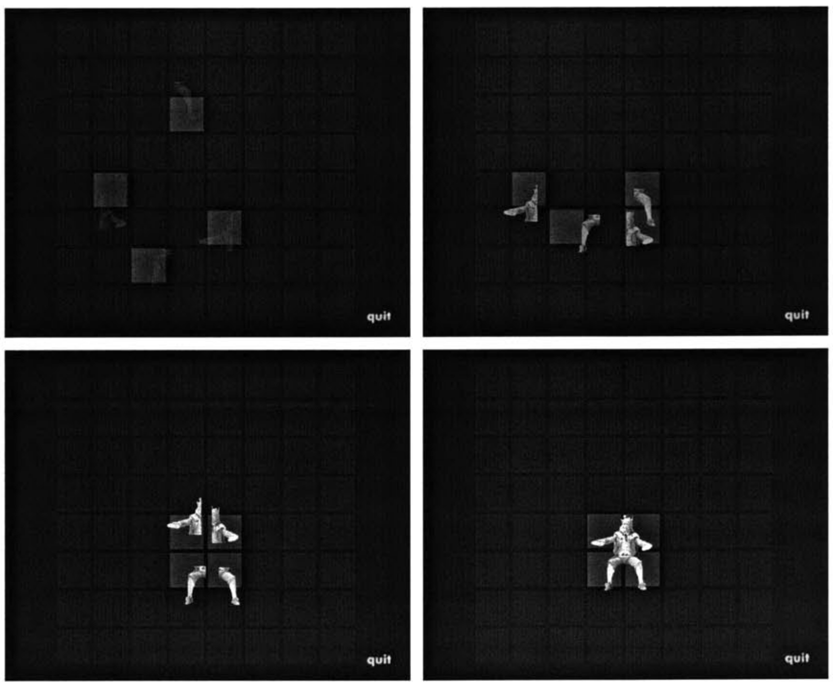
In Wild Leprechaun, there is no interaction, only visual and tactile observation. The visualization maps directly to the haptic feedback provided by the pad, similar in concept to the "What You See Is What You Feel" (WYSIWYF) haptic displays developed at Carnegie Mellon University [YOK 96]. The program served as a proof of concept for the liquid haptic feedback device.

Wild Leprechaun was informally tested to understand its effectiveness on different people. The opinions of those using the program were quite similar. They felt the heartbeat was easy to localize with the hand. The more random movements of the four pieces moving independently could be felt en masse, but individual pieces could not be tracked or separated ("bits and pieces", "hand can't sense four different parts"). The tactile sensation made much more sense while watching the simultaneous animation on the screen. But after attending visually and tactually to the simulation for a few minutes, people generally could generally use the tactile



When the leprechaun is magically darting about, the person feels four distinct vibrations moving along the underside of the hand. When the he comes together and rests, the vibrations are replaced with one heartbeat.

pad alone if the screen was turned off to figure where the leprechaun had moved to and whether he had split into parts or settled in one place ("I feel the movement", "He's right here"). When split apart, the leprechaun pieces were represented by one electromagnet each vibrating with a 50% duty cycle at approximately 30-hertz. This was easily felt and was described as "tickling", "twitching", and "spatial buzzing".



Four frames from Wild Leprechaun show the leprechaun materializing, which can be felt on the Liquid Haptics output pad.

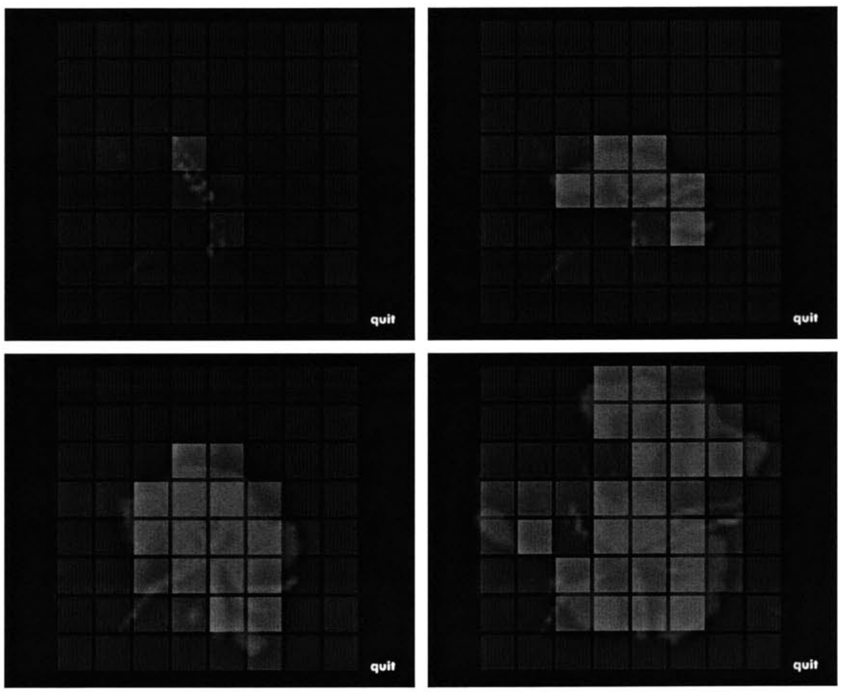
This program served as a good haptic litmus test to indicate the effectiveness of the force feedback technique because it is very simple to understand and follow, and it is completely passive in that no action is required of the person using the pad. The heartbeat was compelling as this is a sensation tactually known by most people, and the metaphor is easily understood and engaging.

4.5 Liquid Touch

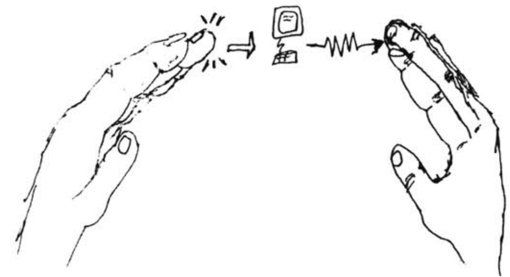
Inspired by the inTouch work in the lab by Scott Brave [BRA 97], a collaborative environment between the input liquid haptic pad and the output liquid haptic pad was created. This application takes the pressure from the input pad and transmits it to the output pad where it is translated into spatial vibration. Thus, the computer serves as a communication channel linking two purely haptic spaces.

Liquid Touch is interesting because the computer and display are taking a background role. The computer's role is simply to route haptic information from one pad onto another, creating a haptic conversation between two people. Unlike inTouch, Liquid Touch does not use the metaphor is of a "shared object" since the pressure is remapped to vibration and the communication is not bi-directional. However the size and alignment of the input and output spaces are congruent, which enhances the sensation of actually touching another person. Since the communication is also very efficient (10 bytes per frame uncompressed), it would also be an excellent contender for haptic communication over long distances through slow bandwidth channels.

The unobtrusiveness of the computer made Liquid Touch very exciting. I found the mapping from one liquid haptics pad to another much more straightforward than mapping from information that was primarily visual into tactile. The usefulness of the input pad as a low overhead spatial device was also reasserted as Liquid Touch became the easiest way to activate specific regions of the output pad in further perceptual tests and debugging of the output device.

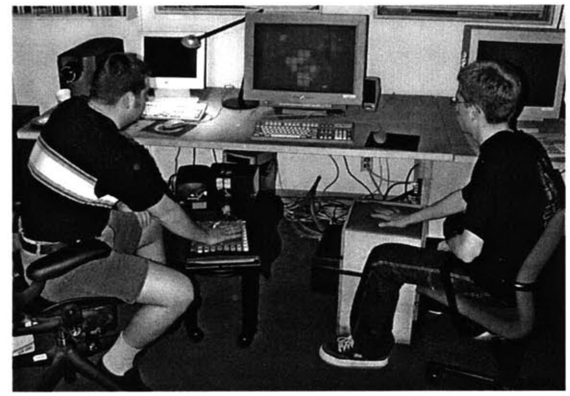


Four frames from Liquid Touch show pressure information as a person's hand is placed on the input pad. The highlighted square cells show active regions, which are felt as vibration on the Liquid Haptics output pad.



In Liquid Touch, the computer takes a the background role of simply translating the pressure information from one person to spatially congruent vibration for another person on a separate pad.

In informal user testing, people felt a strong communication bond through the liquid haptic devices. Both spatial and pressure information was tightly coupled from the input to the output pad. When people pushed harder on the input pad, people felt more activation on the output pad ("definitely feel how big it is", "feels like a lighter touch"). People could follow the other person's hand as it traveled over the pad ("I can track it", "over there, and now down"). Similar to Wild Leprechaun, the apparent tactile registration was accentuated by the visualization. But even blindfolded, a person could describe a new hand gesture by another person using purely tactile information, for example a counterclockwise swirling motion.



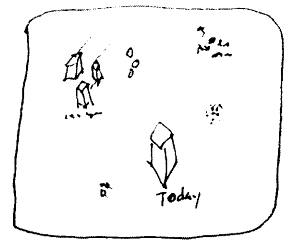
Two people use the Liquid Touch system to communicate haptically.

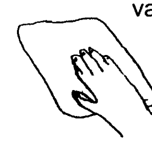
4.6 Uncoded Possibilities

A masters program is nasty, brutish, and short, and unfortunately not all interesting coding ideas were realized for this thesis. Generally the applications that involved complicated subsystems were dropped because the same ideas could be presented in the simpler systems above. But it is important to discuss some of these as mental mock-ups in order to see how these existing liquid haptic devices could be used in applications with larger scope.

The program in WorseWindows hinted at the utility of recognizing gestures. An ability to recognize gestures could allow the person to change modes so that hand motions are now interpreted as perspective changes and not object operations. Because of the rich data available to the computer, it would be possible to implement a family of possible gestures with more sophisticated vision algorithms. Even a small number of gestures would be enough for a huge set of possible interactions if a grammar of gesture meanings were constructed, similar to the work done in the Sketch system [Zel 96].

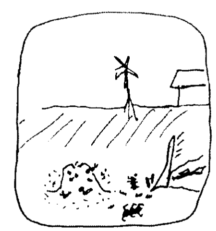
The world is full of complex multidimensional data. It would make sense that a high bandwidth interface would be appropriate for navigating or manipulating a data rich environment. One exciting area is in multidimensional scaling, which maps distances in *m* dimensional space to distances in *n* dimension space minimizing global error [BOR 97]. Using this technique, complex high dimensional data could be mapped onto the two dimensional communication area of the device. Small data sets may even be manipulatable in real time [KLO 97] so that a person can explore an eight dimensional space by literally turning it in his hand. With a force feedback liquid haptic device, the stress value inherent in each data point in the lossy mapping could be mapped to vibration, which would point out instabilities or error in the visualization.





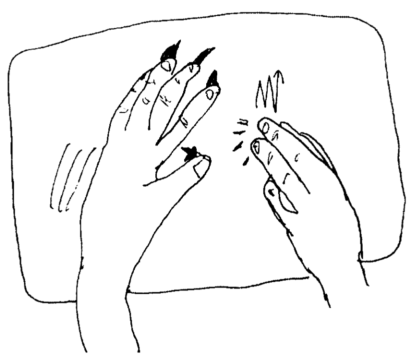
With a liquid haptic pad, one could get a better "feel" for multidimensional data by turning it to look at from different perspectives and feel the stress contained in each data point which represents variable error that occurs in scaling each point into two dimensions.

The liquid haptic devices are naturally geometric and well suited to two-dimensional tasks such a finger-painting. Other two dimensional systems that might benefit from multipoint analog control include games, page layout using simultaneous object translation + scaling + rotation, landscape architecture sketches, and sculpture. Even two-dimensional systems not generally controlled in massive parallelism like cellular automata could be coarsely controlled or felt in real time.



A person with a liquid haptic device plays a futuristic game which simulates anthill colony by computing each ant's behavior in parallel. Here, the person sees the reaction of the ants after plowing through the anthill with varing pressure and direction, and can feel the ants scurring away under his plow-hand — which may give clues as to how to form the next attack.

Other applications are not geometrical, but could benefit from liquid haptic devices because of the high bandwidth available for real-time control. An example of this would be in creating a musical instrument. Generally musical instruments are high bandwidth devices that translate a person's body movements into very specific sounds with bending tones, variable attack strength, and a wide dynamic range. With a high bandwidth device, the computer-based music does not have to be as constrained since there is enough information to play a rich tambral and dynamic mapping in real time.



A futuristic musical instrument based on a liquid haptic pad lets a person control bass notes with the backside of the hand, bend other voices with the fingers, and use the other hand to control rhythm and attack.

5 Discussion

The work with Liquid Haptics can be evaluated in several contexts. It is valid to examine the engineering work necessary to construct the pad and examine its relationship to other human computer interfaces. It is valid to examine the Liquid Haptic devices as an extension of the work in the Aesthetics and Computation group in exploring the medium of computation as a means for mediating human expression. Perhaps it is valid to examine the Liquid Haptics devices themselves as computer art. Obviously we must first establish a framework for understanding the purpose of this work and moving between these perspectives.

5.1 Discussing the Discussion

People curiously tend to avoid real discussions centered on certain topics. When jazz musician Fats Waller was once approached by an earnest young lady with the inquiry, "Mr. Waller, what is jazz?" He replied "Ma'am if you've got to ask, you'll never know!"

Similar to Fats' reluctance to define his music is a general tendency of people to avoid examining humor. Most people find the act of explaining a joke to another person distasteful, and in no circumstance does explaining the joke make it any more funny.

There is often a similar adversity to discussing art, especially in dissecting specific art works to examine what makes them beautiful or meaningful.

I believe these disparate topics share an important quality. The jazz music of Fats Waller, a funny joke from a friend, and a moving work of art are all intensely personal experiences. An attempt to analyze a subject of such a personal nature has the quality of an attack on that person's identity.

This resistance and resentment to dialogue also carries over to aesthetics, which concerns the nature of beauty. But it is in fact important to examine our personal reactions to our technology, so this negative impulse must be overcome. Dialogue examining what we find important and meaningful gives us a better understanding of how to build artifacts that we enjoy instead of living in a staid environment. By not feigning interest in that which we do not understand and keeping honesty above pride, we can cut through misconceptions and understand what is of personal value. If we understand what improves our quality of day-to-day life we can design things that enrich our existence.

I not only believe that beauty can be engineered, I believe it should be.

5.2 Computational Art

It is interesting to examine liquid haptics in relation to computational art. Computational art is first defined, then explored by looking at a few examples of computational artists.

5.2.1 Definitions

> *All the days of my life I have seen nothing that has gladdened my heart so much as these things, for I saw amongst them the wonderful works of art, and I marveled at the subtle ingenia of men in foreign lands. Indeed, I cannot express all that I thought there.*
> Albrecht Dürer, journal entry, 1520. [BAR 93]

> *"I really liked it, but what did it mean?" For some reason, they thought that what had happened should be reduced immediately to words. In fact, people have a tendency to accept events in terms of the words that they will use to describe them. Therefore, there is a place for a medium that can resist interpretation.*
> Myron Kruger explaining how artistic analysis discourages the artistic experience, 1990. [KRU 90]

Art is exclusively the domain of man. The artist conveys a vision in tangible form, and presents concepts and ideas directly. Artistic expression conveys the human experience phenomenally, sometimes circumventing analysis. Artistic response is a personal experience of inspiration, boredom, meditation, or confusion depending on the expectations of the viewer and the mastery of the presenter. The artist presents his experience, perceptions, and reality through his work. An artistic work serves no greater purpose. It is an end in itself, to be appreciated by others.

Computational art is indeed art. But computational art is more than simply art engineered with or presented on a computer. The computer is its own medium. The medium is based on the logic unique to computational processes.

A person who paints a building is not necessarily an architect. A person who sculpts elaborate characters is not necessarily a writer. A person who skillfully choreographs dancers is not necessarily a musician. And a person who uses computers to create any work that is not inherently of computational nature, including beautiful traditional artwork, is not necessarily a computational artist. To call this person a computational artist is to confuse the artistic intent with the enabling technology.

This understanding of computational art is relatively new, and has not been universally accepted. For example, examine the attitude toward computational process that Lillian Schwartz presents in her *Computer Artist's Handbook* [SCH 92]:

> *Over the years, when I encountered technological roadblocks, I eased myself into learning what was necessary to overcome the obstacle. Sometimes, I delved into programming; sometimes, I learned those complex words that would convey my thoughts a scientist with whom I was collaborating. Anything more, and the artist can become locked into the computer. Knowing a programming language and then writing a program using that language can lead to creative isolation for the artist, because the birth of the program (often followed by the need to tinker with it over the years) displaces the artistic act. The program becomes the artwork, and the fontanelle through which creativity has previously surged unimpeded becomes impenetrably blocked.*
> - Chapter 1: Developing an Approach, p15-16.

It should be obvious that Schwartz is not excited about exploring the computational medium on its own terms. Her disdain and misunderstanding of computer programming shows she is not interested in exploring the computational medium, but rather exploiting its strength to ultimately produce other artifacts. In this vein, her computer is a flexible and powerful tool for the traditional artist, much like the introduction of oil paints to a painter who had previously known only watercolor.

For the computational artist, the idea that programming leads to creative isolation is absurd. Programming is designing a computational process at the finest granularity. For many computational artists, this is the ultimate act of creation. With the logic of computers, an artist can express himself in ways never before available. The ability to interact and behave autonomously adds a new dimension to what was before possible, similar to the evolution from photography to moving film, but more profound.

Myron Kruger states it thusly [KRU 90]:

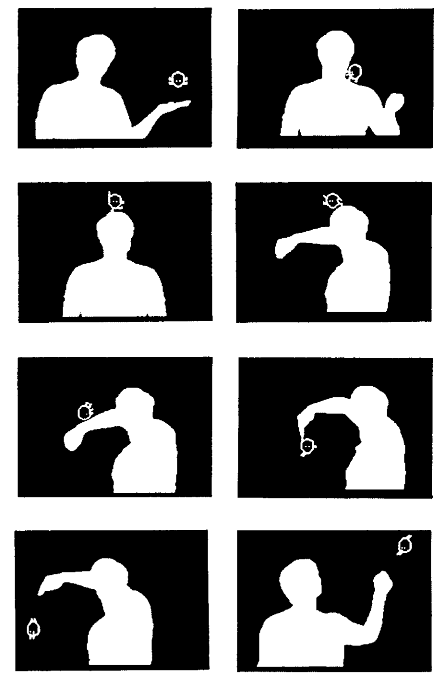
> *Much of the art that has produced with the computer could have produced by other means, albeit tediously. True computer art would be impossible without a computer. One essential aspect of the computer is that it can assimilate information and make decisions in real time. In the past 20 years, artists have attempted to create an art form dependent on this computer capability. Although there are only a few such works, they point to a major new thrust in both technology and art.*

As a better understanding of the nature of computers grows in our culture, so does the number of people able to appreciate this new form of computational art. Inevitably, creative acts of logic design will be appreciated in their own right as computational art: a creative act that mixes with, but is not dependent on, other artistic disciplines.

5.2.2 Examples

The following is a sampling of a few important computational artists.

Myron Kruger has been creating new forms of human-machine interaction for the past thirty years. His main work was exploring a medium he calls Artificial Reality, mainly though his VIDEOPLACE system [KRU 90]. VIDEOPLACE is a system that uses camera vision to interact with people, for example watching posture or movement, and presenting the participant with a computer controlled graphic world on a large projection screen. The exploration of this medium consisted of writing programs that used the computer's knowledge of the person's activities to create real time interactive experiences. Often these programs involved projecting the person into an artificial graphic world by composting their silhouette with computer controller graphics. Different interactions included painting with body motions, shaping spline curves with several fingers, interacting with another person in a virtual space with strange physics, or playing with an artificial creature that reacts to your intrusion to his artificial world.



A VIDEOPLACE sequence with CRITTER, a playful sprite with an artificial personality.

5 Discussion

Harold Cohen created a large autonomous program that is capable of creating expressive line drawings of people and objects [MCC 91]. This program is called Aaron as Cohen considers it an artist in its own right, capable of endless creative drawings. Aaron was the first example of meta-art: creating a work of art that itself is capable of creating art. Aaron uses artificial intelligence to embody the artistic sensibilities that Cohen has programmed into him. Thus, Cohen has used the unique abilities of computation along with careful analysis of his own creative process to create an infinite variety of drawings. In the process he also forces many people to reconsider creativity and artistic intent as the act of artistic creation is separated from the final artwork through a machine.

An image generated by Harold Cohen's artifical artist program called Aaron.

Karl Sims has developed the process of aesthetic artificial evolution to create dynamic forms of higher quality and complexity than either the person or the computer could develop separately [SIM 97]. The process of interactive evolution becomes a collaboration between a person and the computer. Sims uses the power of the computer to create an endless variety of shapes closely related to the aesthetic preference of a human operator. By iterating this process, a person's selection becomes a creative process of selective breeding to create an end result that is not constrained by the limits of human design or understanding.

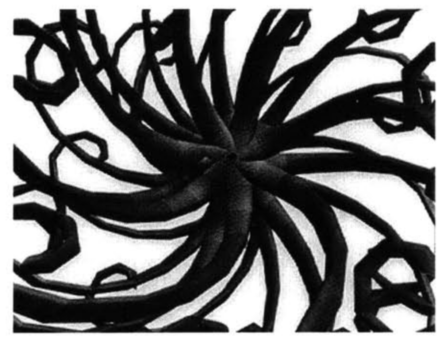


Image of one of Sims' evolved virtual organisms.

Each of three examples is unified by a human act of creation within computational media. Each of these three has also expanded the domain of computational art by introducing new creative concepts and processes. Without the ability to sculpt logic, each of these accomplishments would not be possible.

5.2.3 Personal Work

The author's own work in computational art has been developing over the past five years.

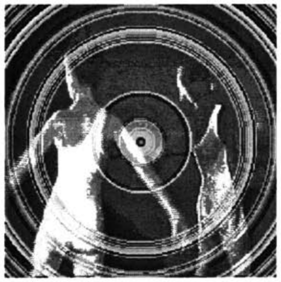
In 1993, I worked with Myron Kruger on *Small Planet*, an art installation for SIGGRAPH 1993. As an intern and apprentice for Kruger, I worked for 5 months designing software that used the perception abilities of his VIDEOPLACE environment. Kruger envisioned Small Planet as a way to allow people to fly over a small planet using natural arm motions, get first hand experience of navigating the sphere, and share this experience by networking two environments so two people are flying in the same shared space. For the Small Planet project, I developed the geometry and textures for the graphic planet, the code that interpreted hand position from Kruger's hardware as a direction of flight, and the algorithm that controlled the projected view window to effectively fly them through the space.

A scene from the *Small Planet* installation. Here the author is flying over a graphic planet using natural arm motions.

This was my first experience working on a computer program that would later be used by thousands of other people. It was also my first experience designing a responsive interactive environment.

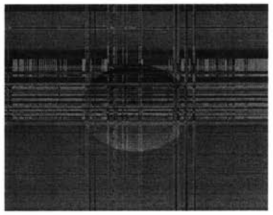
Picture of the author alongside one of his larger works in the art show that accompanied the University of Georgia art class. Only partially visible here, he artwork is a large image of two hands that change when viewed from different angles. This work was made up of dozens of bulging strips of paper printed from postscript files that were themselves generated from C code.

Later in 1993 I returned to school and entered an art course centered around the computer. Though the class was designed primarily a means of teaching artists common computer graphics tools, I used it to explore the possibilities of using the computer to create static, non-interactive work that was still true to the medium of computation. The class was also important because it was my first introduction to many basic artistic concepts. Through this class I was able to consider how to use the computer as a extension of my creative abilities.



A frame of animation from one of several dance performances where live video of dancers was processed by the computer and projected back into the dance space during a performance.

The following year I collaborated with a local modern dance company to create performances that integrated computers into dance performances. A combination of cameras tracking the live dancers and taped footage was processed on my computer and projected during portions of the performance. The computer was one of many components of the dance, and would alternatively be the focus of attention or a backdrop to the dancers. Live performance for a passive audience and artistic collaboration were new skills that this work introduced.



Two images taken from the SIGGRAPH 1995 Injection exhibit

In 1995 I exhibited my own interactive installation called *Injection* at the SIGGRAPH Interactive Entertainment exhibit. Injection used genetic algorithms and cameras to create complete interactions that were experienced on a large projection screen. The interactions presented themselves as animations on the screen that changed as people interacted with the system. The exhibit consisted both of the individual computer interactions and the breeding process that spawned them. This was my first large independent effort, and my first time seeing my own ideas mature into a work experience by many people.

5.2.4 Liquid Haptics as Computational Art

With this framework for understanding computational art along with my own artistic background established, we return to the question of what the relationship is between the current liquid haptics work and computational art.

5 Discussion

I consider liquid haptics not as an example of computational art in and of itself. Rather, it is an enabler of better computational art because it is engaging, natural to use, and by its very nature questions how people interact with computers. To say it is computational art would be to recognize it as important expression in its own right, which it is not. This is a rational and personal decision by the creator of the device, and depends entirely on his judgement.

A good example of liquid haptics used within computational art is its incorporation into the Stream of Consciousness collaboration with David Small. Stream of Consciousness is an instance of a computational artwork, the project exists as an artistic end in itself. This project has been accepted into the SIGGRAPH art show, where it will be exhibited in July 1998. The liquid haptic input pad presents a natural interface to the flowing water and words both in complementing the liquid of the actual stream and as an efficient interface to the large two-dimensional graphics area. The liquid haptic pad allows people to interact with the environment in a relaxed state of mind, whereas traditional interfaces including a mouse or keyboard would be more apt to feel like work than meditation.

Other more complete pieces of computational artwork could be constructed using the liquid haptics input or output pads as natural and intriguing interfaces. Separate from their utility at measuring hand data, liquid haptic devices are well suited for interactive installations because they disarm most people of their computer preconceptions by presenting them with a mode that is unfamiliar, causing them to question their relationship to the computer, and perhaps at the same time making them more willing to accept and explore other new computational ideas.

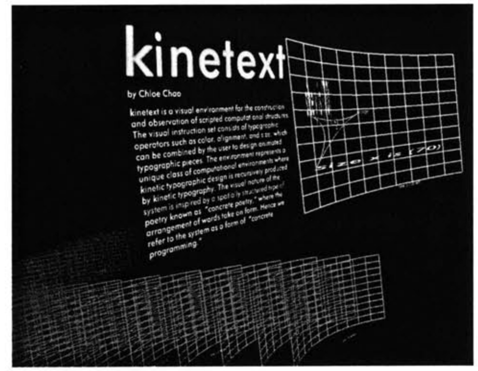
5.3 Aesthetics and Computation

5.3.1 Purpose

The goal of the Aesthetics & Computation Group (ACG) is to establish the use of computation as a means for mediating human expression. We study the expressive aspects of computer-human interface from the viewpoint of traditional visual communication design. We find that the computation itself is a candidate for its own field of design: computational media design. Computational media design involves using the medium of the computer to explore various man-machine interactions in search for those that are engaging and expressive.

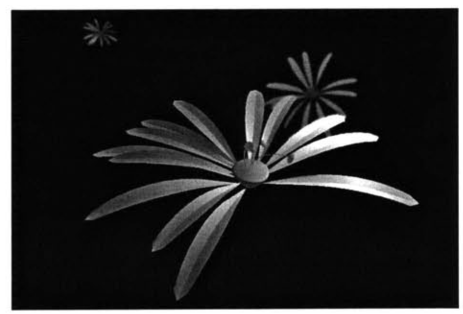
5.3.2 Examples

Work in ACG includes the following projects.



Chloe Chao's Kinetext workspace.

Kinetext is a programming sketchbook environment for animated text design by Chloe Chao. Chloe is creating a visual environment for creating dynamic typographic animations. Chloe's work both explores the process of creation by examining the issues in replacing a programming problem with a visual process and also creates a new and visually stunning workspace.


A visualization from Matt Grenby's Glom.

Glom is a project by Matt Grenby to develop a visual representation for quantitative information. Matt is exploring the issues in mapping a large dynamic data set onto organic shapes so that they can be used and remembered more easily.

Reed Kram is working on a new dynamic pictographic visual language for describing interactive digital performance systems. At the center of this language is a diagrammatic environment called Modulator. The goal is to use Modulator classify existing digital performance spaces into a framework that can then be used for comparison, analysis, and discovering new types of digital performance.

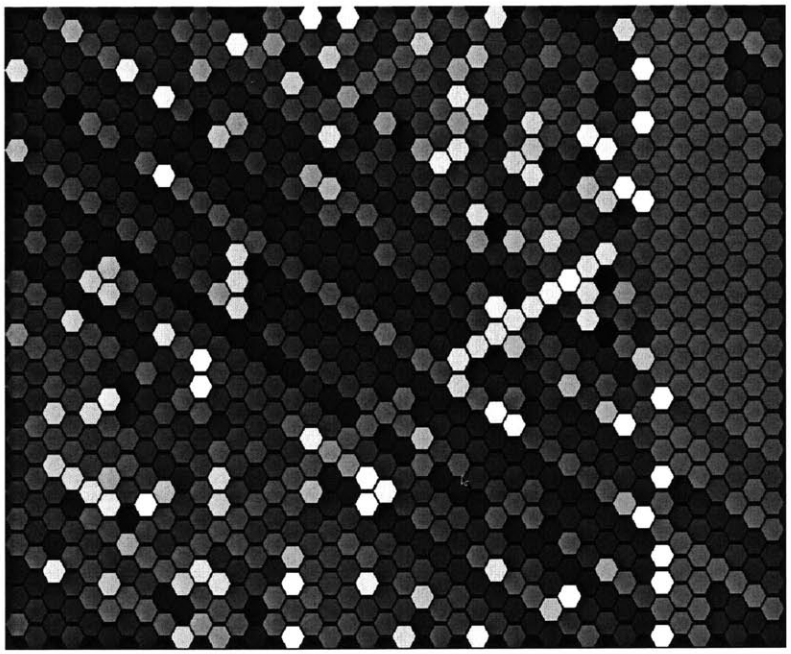
Reed Kram's Modulator environment.

A dynamic representation of colors in Peter Cho's DataPaint.

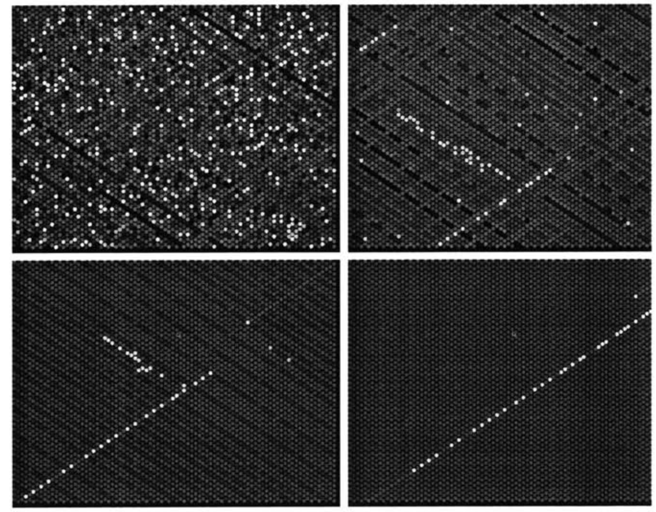
Other work in ACG includes David Small's work Reexamining the Book, in which he investigates the role of the book in today's society and explores how the computer can enhance, augment, or replace the book with online text. Sol is a project by Richard DeVaul to create a programming language that is designed using ideas from set theory in order to more easily explore design forms. Peter Cho's current project is DataPaint, a gesture based system that explores how to create a local graphical structure, and then broadcasts that structure over the network in an appropriately different format.

5.3.3 Personal Work

Upon arriving to the Media Lab, I began working on a visualization of the computational process itself at the lowest level. I began a project called VISC in which I constructed a graphical two dimensional Visual Instruction Set for Computers, similar to the instruction set a computer processor can implement in hardware. A single graphic cell represented each atomic computational instruction, and program flow passed to one of the adjacent cells. VISC allowed multiple simultaneous execution locations, similar to a computer with several parallel processors. Various regular tilings of the plane were examined, including using squares, triangles, and hexagons. Sketches of control structures like loops were designed. Though no compilers or other code building tools were made, the VISC systems worked well enough to display emergent behavior when run on an area of randomly initialized data.

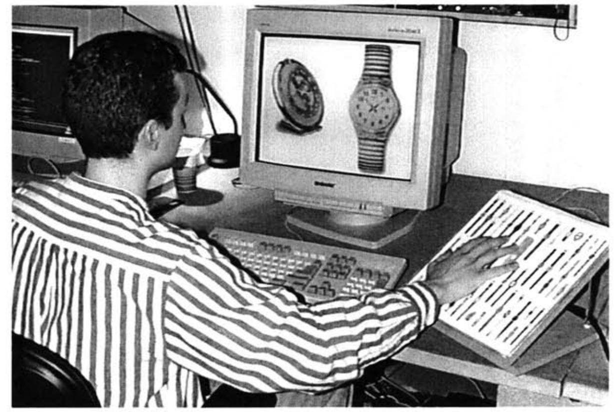


VISC is a visual model of computation in which cells can read and write onto neighboring cells, several parallel threads of execution run at once, and each thread can pass control of execution control to itself or any neighbor on the next clock cycle. Here several dozen light colored cells represent active VISC threads on a hexagonal grid.



Four screenshots of VISC run on a hexagonal grid with randomly initialized data over a period of 8000 cycles shows emergent behavior from initial disorder.

Other early work included designing an interface using the wireless capacative fish sensors developed in the lab [SMI 98]. The interface I designed was called FishPad, and used a north, south, east, west configuration of fish sensors embedded in a one foot square sensor to measure limb position. The primary application developed was using this interface to quickly zoom and pan through large images. Other work including setting up two pads for a two-handed interaction. The concentration was to create a new way of interacting with computational media through an interface responding quickly to arm movements. It was this work that led to tactile pressure experiments, and later to liquid haptics.

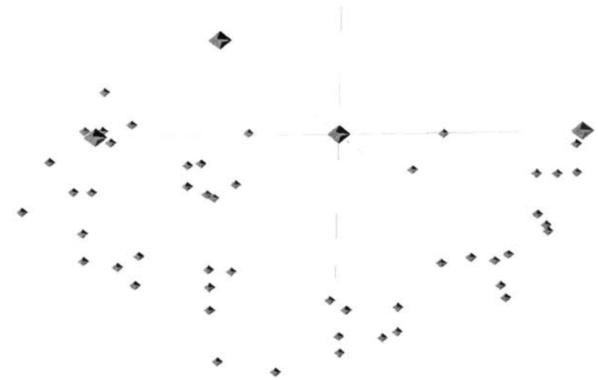


Here the author is using his final FishPad design to browse through an inventory of watches. Moving the hand in three dimensions zooms and pans through the image.

Another interface developed around this time was the Puk system. Puk was a system for collaboration between up to eight people. The system used a large square table at which eight people sat, and each person had three small disks. The computer tracked each person's disks, and a graphic was projected in the center of the table. All disks were tracked simultaneously with one vision system. The system was developed along with Andy Dahley for a final class project. I developed three applications, including an interactive game. This project was challenging because the interaction had to be quick, the graphic had to make sense from any location around the table, and input for each person was limited to the spatial relationship of three points in a plane. But it was very rewarding to create a computer interface that easily allowed eight people to interact in one large space simultaneously.

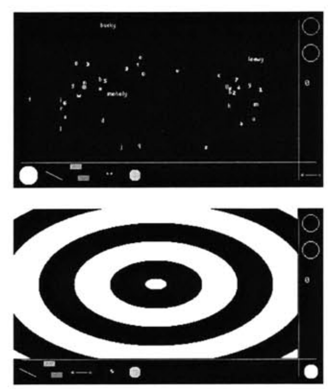
A person moves two disks in the collaborative Puk system.

Swarm was a brief project I worked on as a way of understanding and visualizing how the operating system treats multiple simultaneous threads that have various scheduling priorities. The system was written in OpenGL and consisted of dozens of small particles that emerged from a source location and drifted toward a destination. Each particle was run in software as a separate POSIX thread. As a particle approached the origin of the coordinate system, it would begin to light up, and within a certain distance the scheduling priority of that particle's thread would be increased. It was thought the particles would slowly speed up near the origin due only to the fact that such particles would be getting more timeslices of the system. But as soon as a particle was within a critical range of the origin, what happened instead was that particle would instantly zip past all others. Thus, swarm was able to visually show how very poorly the POSIX thread implementation was on that operating system as the scheduler was using a "winner take all" scheduling algorithm, and the project was subsequently stopped.



A screenshot from swarm shows particles drifting from source to sink, with no particles near the origin because the scheduler makes the region around the origin an unstable state.

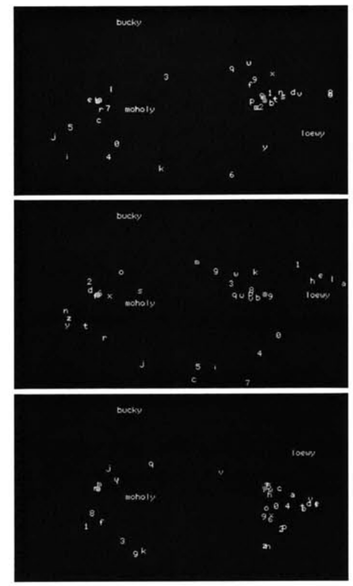
Continuing with software exploration, I worked on a project called Buzz that examined distributed computation, the use of the computer as a shared space, and having executable code dynamically travel from computer to computer through the network. Buzz was programmed in Java and was setup to run on the login screens of computers, so inactive computers could all inhabit a shared space. Buzz was written to be a completely distributed networked application; when the program runs it uses multicast to search for other buzz machines on the network as opposed to relying on one central server. It also builds a local model of active machines, including how far away other machines are on the network as measured by network router hops. It then loads in lightweight BuzzPanels, which are graphic front ends to the system. The BuzzPanels were made to be easy to program. An interface was made to examine which modules are available locally, and



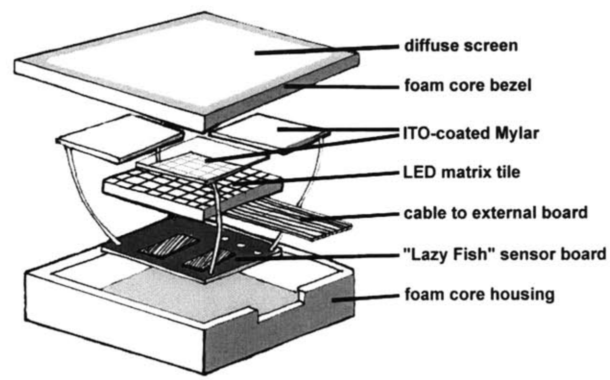
The interface to the local BuzzPanels is brought up at bottom. Available BuzzPanels are represented with an icon. Below, a new BuzzPanel is made active. and the display updates.

any local module can be made active. Active modules can be sent out to all other listening BuzzStations on the network, including the complete current state of the BuzzPanel. For example, a line drawing program could be made active and a new design drawn. Then the drawing program itself plus the newly created drawing could be shipped to all other Buzz stations on the network. BuzzPanels could also use network services to communicate directly to one another through the Buzz system. Several interesting avenues were explored in examining the network as a shared computational space and the dynamic nature of code.

Most recently, I left the desktop computer and investigated programming a small handheld device. The device is called a FishFace, and was developed by Paul Yarin and Josh Smith, both graduate students at the Media Lab [YAR 98]. FishFace uses "lazy fish" sensors, which are derived from the same capacitive sensors used in FishPad, and FishFace has only a eight by eight grid of LEDs as output. FishFace programming presented some severe restrictions on what types of applications could be programmed, both in the display resolution and in the amount of code and data that could be stored in the microcontroller. Within these restrictions, several simple programs were developed. Some simply tracked the hand and updated the pad with a dot or crosshair while others were more sublime - including a low resolution radar screen that continuously swept around the face and lit up dots that were recently tracked as hand positions. Also, the classic arcade game "breakout" was distilled down to its essentials and run on the LED grid with hand movement controlling the virtual paddle. Programming for such a restricted environment was an exercise in essentials and efficiency, and a reminder that computational form does not necessarily mean a desktop computer.



Here an animating BuzzPanel contains the names and mouse locations of other Buzz stations on the network. The floating particles each are labeled with an alphanumeric symbol are not shared across stations. These particles are repelled by a cursor's presence. From top to bottom here, the station "loewy" at far right travels upward, changing the particle paths.



Details of the FishFace, a computational device that can run off of a battery and is rougly four inches square.

Three sequential shots of the FishFace program "Snake", in which a person's hand is tracked and followed by a virtual snake of lights.

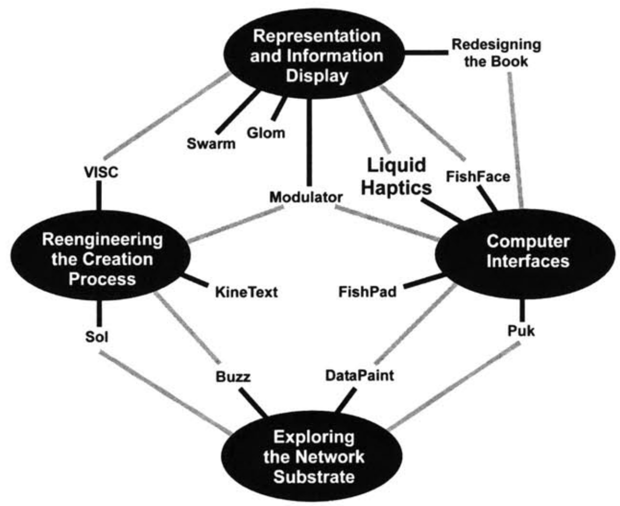
5.3.4 Liquid Haptics as a Contribution to Aesthetics and Computation

The many various projects both by other members of the Aesthetics and Computation group and done by myself throughout my Master's program enrollment have resisted being viewed as converging on one central problem or issue. But some groupings can be made to better see the relations in these explorations into computational design and computer interface.

Projects such as VISC, Sol, and Kinetext explore the creation process itself in an attempt to engineer a more intuitive alternative to scripts or traditional programming. DataPaint and Buzz are interested in the computer network as a substrate and multiple representations across space and time. Glom, Modulator, Reexamining the Book, and Swarm primarily deal with issues of representation and information display. FishPad, Puk, and FishFace investigate computer interfaces by designing new channels of information into computational media.

Thus we have four separate categories of projects. Realistically, many projects touch on more than one area. For example, Sol is being designed to facilitate network exploration as well, and David Small often designs new computer interfaces in order to better navigate his textual worlds. But the categories themselves are a beginning in establishing a classification of Aesthetic and Computation projects.

The work in this thesis is an extension of ACG work done thus far in experimenting with new ways of interfacing with computational processes. The liquid haptic input pad is a new and original means of communicating with computational substrates and facilitates new types of computational expression. The output pad is also a new and original force-feedback tactile device. The output pad also touches on issues of representation and information display, only in a tactual instead of visual manner. Liquid haptics fits well within this young body of work using the computational substrate to develop new engaging and expressive interactions.



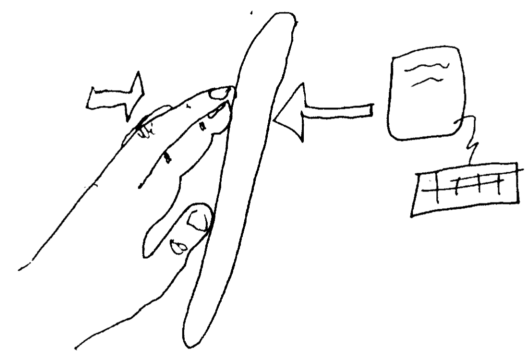
Grouping of projects the author has done as well as main projects in the ACG research area. Black lines represent the primary grouping and grey lines show secondary areas. The liquid haptics work presented in this thesis fits well into this taxonomy.

6. Future Work

Several avenues were explored with existing liquid haptics interfaces. There are several interesting areas of future research in developing liquid haptic devices.

6.1 Liquid Haptic I/O Device

The most obvious and powerful modification to existing devices is to merge the capabilities of the liquid haptic input and output devices into a single I/O device. This fictional device would be capable of measuring detailed information about the placement of the hand and simultaneously could transmit information back to the person by firming or vibrating the liquid. This would present a tightly coupled human computer interface unique to tactile sensation. Such a space would be ideal for real-time control with immediate feedback similar to the GROPE system [BRO 97] or for adding two-way tactile communication to systems like my Liquid Touch.



Ideally, a liquid haptic device would not have to be used for input or output, but could be simultaneously both depending on the context.

The problem with combining the input and output pads is that the input pad uses optics for sensing, which is incompatible with the current output pad design with MR fluid. This is because the MR fluid both has terrible optical properties and sits above an array of electromagnets which almost completely obscures the bladder from a camera's sight below. Thus, one or both designs must change before they can be merged into a single uber-pad.

The most promising technology investigated for this involved replacing the optics of the input pad with acoustic sensing. Carroll Touch Company produces a product known as Guided Wave touch sensors that measure high-frequency acoustic waves travelling through the surface of the sensor [CAR 98]. Their sensors boast over 21,000 points per square inch (148 per linear inch), Z-axis (pressure) capability, a thin 0.04 inch profile, sizes up to 14 inches, and the panels do not require calibration. Because the acoustic waves are not on the surface of the sensor, it is resistant to dirt and scratches, and can even be operated under water. And important for use with the magnetic grid, acoustic sensing should not affect or be affected by strong variable magnetic fields.

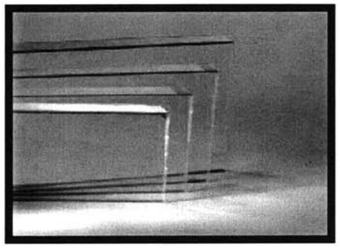


Photo of Guided Wave acoustic sensors from Carrol Touch

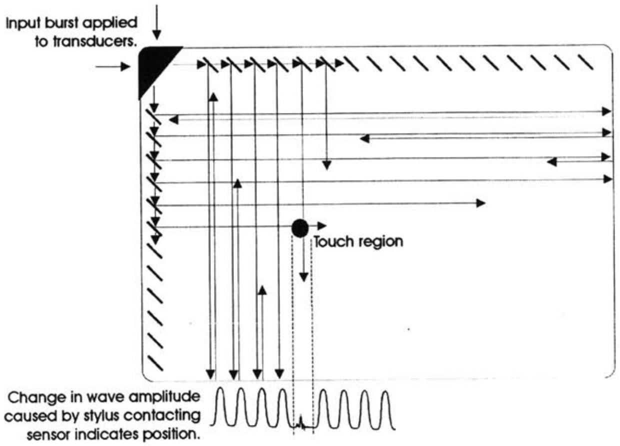


Diagram from Carrol Touch Literature describing the operation of their Guided Wave touch sensors.

Unfortunately, the controller firmware which is responsible for the scheduling and controlling scan operations of the touch system appears to be only capable of reporting the single point of highest pressure contact. It is not clear how pervasive this design decision is in the hardware itself. But it possible that the panel could be modified in firmware to report the pressure over a large area, mimicking the behavior of the current input pad. Or a similar device could be built from scratch with the intention of delivering a pressure map instead of a single point.

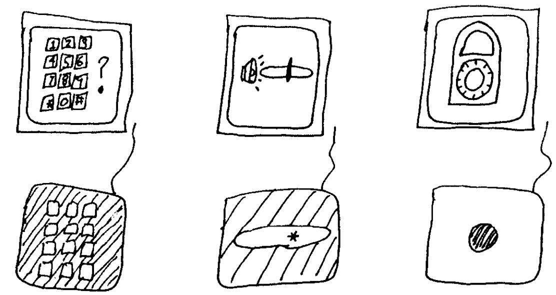
6.2 Morphing Medium and Muscle Memory

Buxton makes a number of interesting observations in developing his taxonomy of input devices [BUX 83]. He sites Baecker [BAE 80] as pointing out that the effectiveness of a particular user interface is often due to the use of a particular device, and that effectiveness will be lost if that device were replaced with some other in the same logical class. He also discusses the important role of "muscle memory" in recalling how to perform tasks, and mentions both his ability to type effectively though incapable of telling someone where a certain key is on his keyboard, and his ability to open a lock whose combination he cannot recite.

There is also a recurring tension between having more input devices versus loading more modes of interaction on existing devices. One extreme says for every function there should be a device to avoid the use of modes which result when a single device must serve for more than one function. The counter stance takes the point of view that the number of devices required need only be in the order of the control bandwidth of the human operator.

An I/O liquid haptics pad could solve many of these problems by using the feedback ability to morph into the appropriate form for a suitable input device. The change would be cosmetic, since the pad would continue to read detailed two dimensional information in any configuration, but the haptic shape of the device would give the person important clues that reduce the cognitive load of using the pad. This approach uses one high bandwidth device that can change shape to appear as appropriately different devices depending on the function to be performed.

For example, if a phone number needed to be entered, the familiar graphic of a phone touchpad could appear, and the liquid haptics pad could morph appropriately to feel like a three by four table of squares. This would allow someone to push on the squares while minimally attending to the pad. Similarly, if someone wanted to adjust the audio volume of the computer, the pad could morph into a familiar slider with a vibrating point to indicate the level, and the person could slide the virtual knob while attending to other activities. And if Buxton cannot remember the combination to his locked valuables, his liquid haptics pad can morph into a lock shape, recognize the twisting gesture on the pad, and update the on screen graphic of his lock accordingly to accommodate his muscle memories.

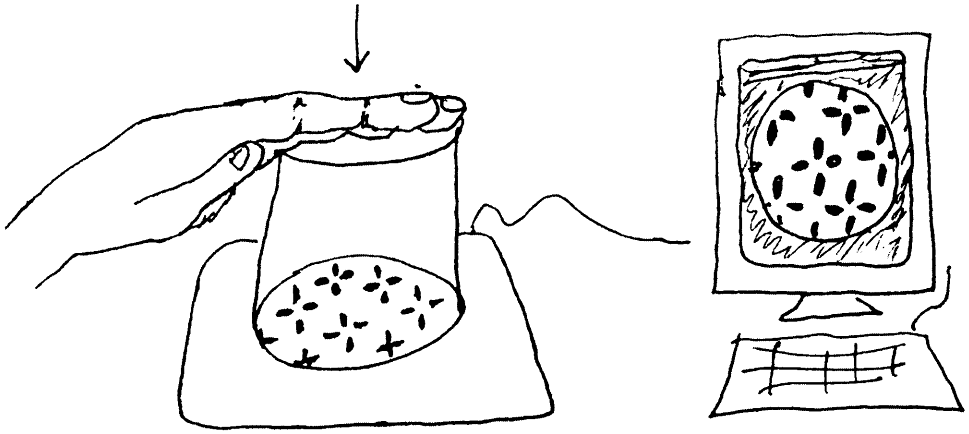


Three sketches of how a liquid haptics I/O pad could morph into various configurations depending on the input requested. At left, the pad hardens except for the area consisting of 12 square pads, mimicking the feel of a touch tone phone for easy data entry. At center, a slider bar is tactualized by having the pad stiffen except for an elongated area representing the input space, and a vibrating point to represent the current setting. At right, a circular region on the liquid haptics pad stiffens to represent the cylinder of a lock, and twisting motions are recognized to enhance the virtual sensation of opening a lock. All three sketches represent interactions with lower cognitive overhead because the shape of the input space both restricts motion and triggers muscle memories.

6.2 Very Accurate Input

Most of my work with the liquid haptic input pad concentrated on using an uncalibrated pad to do tasks that were not necessarily precise in nature. More attention to careful pressure calibration in the Haptic Lens work has shown that the platform is also fit for processes requiring very high fidelity. Using elastomers in place of liquids potentially increases the accuracy of the device, and the elastomers have less apparent latency, do not generate standing waves, and have a variety of visco-elastic properties which could be useful in modeling different types of applications.

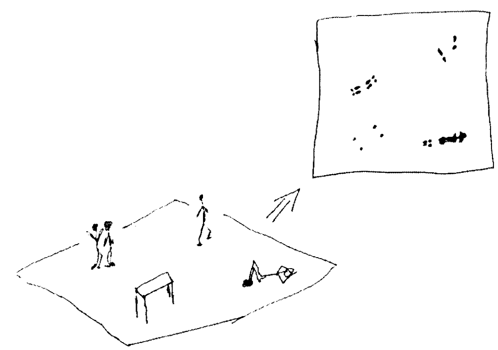
High precision haptic input would greatly increase the usability of the device, including making it more practical for more exacting tasks such as medical applications. Higher quality input space would not only be able to read gross features, but also bumps and textures. Such a rich space could be used not only as an input space, but also as a data window that could translate the shape and texture of everyday objects into the digital realm. Objects to be modeled could simply be pressed on the pad, a texture or a simple shape needed for a sketch could be generated by items lying around the computer, or a record of a child's growth could be made by pressing her shoes on the pad at periodic intervals.



Using a liquid haptic input pad as a data window lets a person easily capture and use everyday shapes and textures for the digital world.

6.3 Larger Input

Work with the micro-bladder used for travelling introduced new issues in scaling the liquid haptic input device. Mentally extrapolating upwards, we would expect that large waterbed sized bladders would require liquids that were much more transparent in order to measure pressure through a larger cross section. Though larger bladders were not built, huge liquid haptic input areas would be an interesting and relatively cheap means of measuring an entire body or floor space. Replacing the pen camera with a camera with a wide-angle lens would greatly reduce the space needed below the bladder, which is important for bladders with larger area. For areas too large or areas without space below, acoustic or other

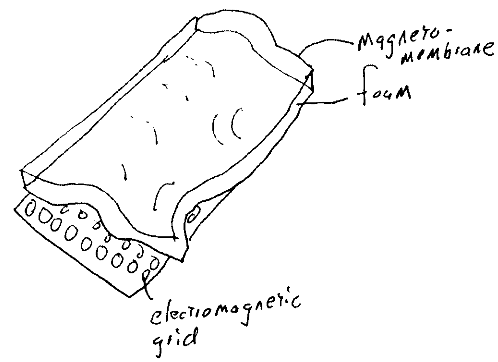


A huge haptic floor could conceivably capture the pressure information from large areas.

types of sensing could be used. With larger areas one can imagine creating an input space out of a chair, waterbed, or even a water-carpet over a large floor space.

6.4 New Output Devices

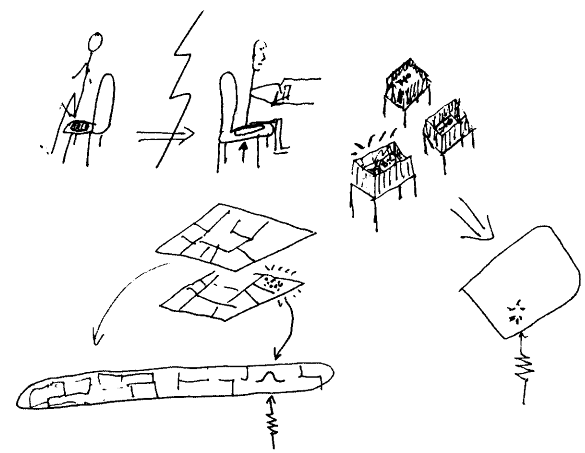
In addition to the MR fluid, Lord Corporation also produces a magnetic elastomer. As a material, it has a peak permeability of about 5 and a saturation magnetization of about 0.7-0.9 Tesla. By layering such a material over an elastomer or thin foam, and placing this pad on top of the existing magnetic grid, a new type of force feedback device is created. This device would be a magnetic-based, real-time morphing relief surface. This is similar to the liquid-based output pad in spirit, but quite different in appearance as the actual surface of the pad would morph as magnetic fields were applied. Existing software and control hardware for the liquid haptic output pad could be used as it currently exists; this new pad would simply be an alternative to the MR bladder. But the morphing would feel very different as the surface is not changing viscosity but moving laterally, potentially applying a kinesthetic force to the hand which the MR fluid is not capable of duplicating. And the dynamic morphing of the magnetic elastomer could also be observed visually as well, unlike the existing liquid output pad.



Sketch of a future foam haptic output device based on a magneto-membrane instead of magneto-liquid; this device could visibly change form by exciting the membrane with the electromagnetic grid.

6.5 100% Pure Liquid Haptics

With bi-directional liquid haptic devices, larger and better input devices in a variety of locations, new types of liquid haptic output devices, and a pervasive network, the number of radically new applications is limited only by the imagination of the programmer designing the mapping between devices. An office could have a low-resolution liquid haptic device in the seats such that the firmness of the chair depended on how many other people were also sitting in their offices, similar to if they were all sitting on one waterbed. A wrist rest on a workstation could be mapped to the floor space of an entire building. A bump in the wrist rest would tell the person typing that somewhere in the building a large number of people were congregating, and the bump's dispersal would correspond to people dispersing from the room. A liquid haptic steering wheel would allow a person to play a tune through the horn, and the wheel could vibrate if the person was approaching an obstacle too quickly or inform the driver which way to turn. Liquid haptics devices in cribs could all be mapped to an output pad so that someone could quickly feel if the infants were all sleeping. The pad could also be mapped back to the cribs so that a person could remotely rock the noisy baby to sleep, and the cribs could simultaneously be mapped to each other so that at other times the children could play with each other. The number of viable applications is limited only to the number of situations where tactile sensation is not obtrusive and there is a sufficient information gradient.



Three sketches of possible applications in a world full of liquid haptic devices. At top left are two of many chairs virtually connected such that sitting on one chair makes the others firmer. In the sketch at bottom left, the floor plan of a building is mapped onto a wrist rest and a lump could represent an area of high pressure, possibly a large gathering. At right, a liquid haptic pad is mapped onto several cribs, and a vibration highlights a thrashing, noisy baby, which could then be gently rocked to sleep through the pad.

7 Conclusion

Liquid Haptics has been introduced as a new family of interface devices. Implementation details have been provided for an liquid haptics input device as well as a separate liquid haptics output device.

Applications have been developed for both devices to examine how people can use these devices in interact with computational media. The input pad has been shown to be a natural way to communicate high bandwidth spatial information to the computer with the hands. The output pad has been shown to provide a low-resolution tactile display composed of no moving parts. Both devices have strengths that make them viable interfaces for certain computational tasks. Future work was suggested for liquid haptic devices of even greater utility.

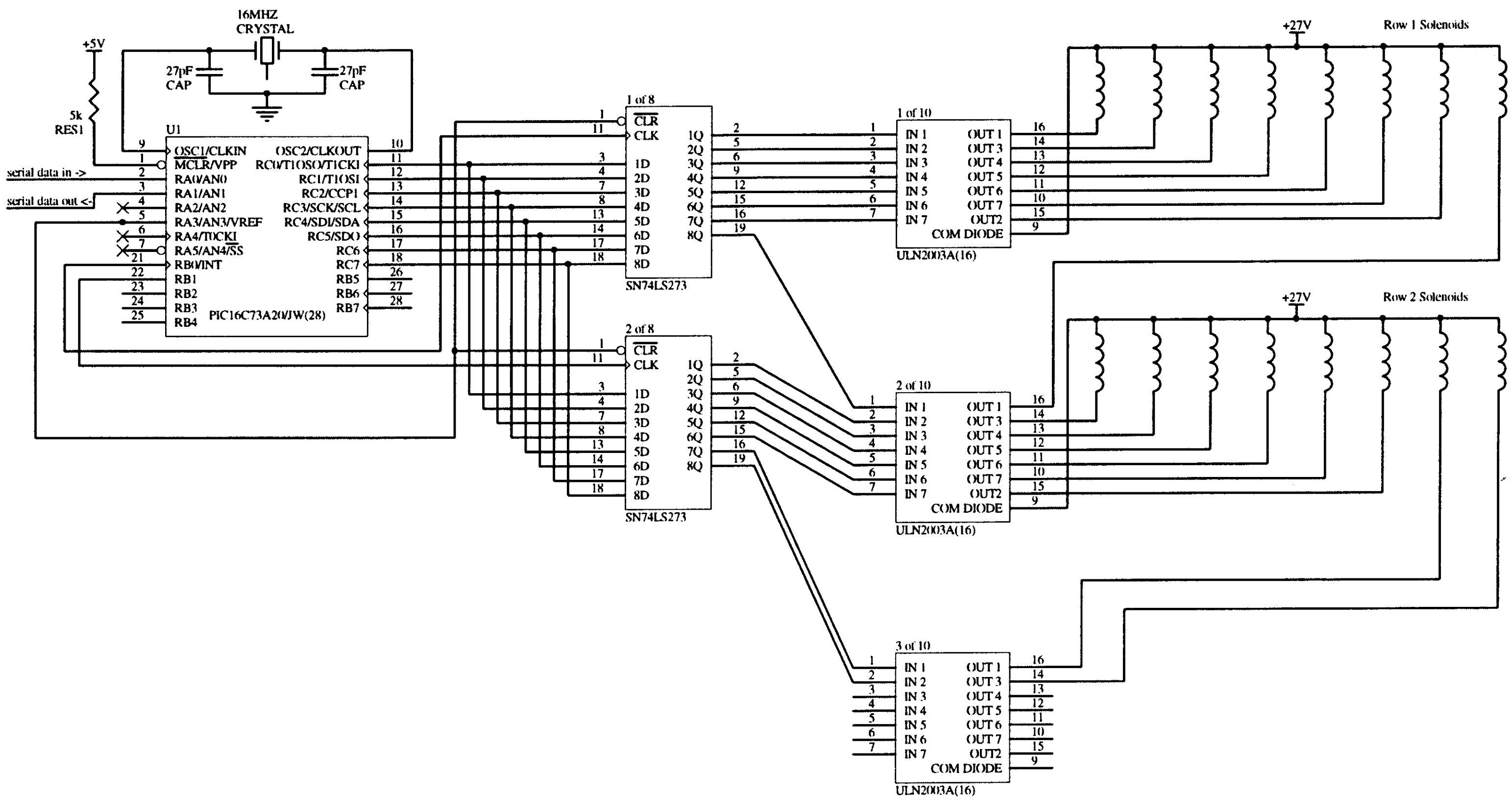
Further evaluation of the liquid haptics project was given, putting this project in context with computational art and the other work in the Aesthetics and Computation research group at the media lab. A complete framework for understanding this and other work was created as a means of establishing the significance of this work within and outside of my research area.

Liquid haptics has been shown to be an efficient and engaging new way of communicating with the medium of computation.

Appendix A: Implementation details

Output Pad Wiring Diagram

The following diagram shows the wiring of the first two rows of the liquid haptics output pad. The hardware and diagram were designed by Paul Yarin.



Output Pad PIC Code

While it is not my intention to list all of the source code used for this project, the C code that was used in the PIC chip is short and intimately tied to the output pad hardware.

The three serial commands decoded by the following program are:

Qxynnn

Summary: set the cell at (x,y) to vibrate with a period of nnn

x: single digit ascii number between '0' and '9' inclusive representing the column of the cell

y: single digit ascii number between '0' and '9' inclusive representing the row of the cell

nnn: three consectutive ascii numbers between '0' and '9' inclusive representing the vibration rate. This number means that every 'nnn' cycles through the event loop, it will reverse the electromagnet at (x,y). Two special values are '000' for turning the cell off and '255' for turning the cell on completely. Values outside this range are invalid.

Rb

Summary: set the overall vibration rate to b

b: the overall vibration rate. This is used with the S command.

Sbbbbbbbb

Summary: reset the entire pad to the binary string bbbbbbbb

bbbbbbbb: each binary digit b represents an entire row of the output pad, which one bit per column. If the bit for that column is on, then the rate on that cell is set to the overall vibration rate, which is set independently with the 'R' command. Thus, these eight bytes completely specify a new state for all cells.

Code:

```
//FOR PIC 16C76!!!

#include "..\include\16c76.h"
#include "..\include\stdio.h"

#define TX    PIN_C6 //PIC pin 17to MAX233 pin 2
#define RX    PIN_C7 //PIC pin 18to MAX233 pin 3

#define COL0PIN_C0 //PIC pin 11
#define COL1PIN_C1 //PIC pin 12
#define COL2PIN_C2 //PIC pin 13
#define COL3PIN_C3 //PIC pin 14
#define COL4PIN_C4 //PIC pin 15
#define COL5PIN_C5 //PIC pin 16
#define COL6PIN_A0 //PIC pin 2to MAX233 pin 2
#define COL7PIN_A1 //PIC pin 3to MAX233 pin 3

#define LATCH_CLR PIN_A3//PIC pin 5 to LS273 pin 1 (latch reset)
#define ON  1
#define OFF 0
#define all_out0
#define six_out0x3f
#define all_in0xff

#byte port_b = 6 //Rows
#fuses HS,NOWDT,NOPROTECT,PUT,NOBROWNOUT
#use delay(clock=16000000)
#use rs232 (Baud=19200, Parity=N, Xmit=TX, Rcv=RX)

unsigned long cycle = 0;
int buffer[8]={0,0,0,0,0,0,0,0};//setup buffer
int table[8][8] = { { 0, 0, 0, 0, 0, 0, 0, 0} ,
            { 0, 0, 0, 0, 0, 0, 0, 0} ,
            { 0, 0, 0, 0, 0, 0, 0, 0} ,
            { 0, 0, 0, 0, 0, 0, 0, 0} ,
            { 0, 0, 0, 0, 0, 0, 0, 0} ,
            { 0, 0, 0, 0, 0, 0, 0, 0} ,
```

```
        { 0, 0, 0, 0, 0, 0, 0, 0} ,
        { 0, 0, 0, 0, 0, 0, 0, 0} };

//INITIALIZE
//set I/O lines, test LED's, test serial lines
void initialize() {
    set_tris_a(0x02);// 0000 0010, A0=serial TX, A1=serial RX
    set_tris_b(all_out);// 0000 0000, all LED row pins set as outputs
    delay_ms(500);
    output_low(LATCH_CLR);//resets all latches to low state
    delay_ms(500);
    output_high(LATCH_CLR);
    port_b = 0;
    printf("\r\nI put the phat in emphatic!\r\n");
}

//UPDATE_PAD
//refreshes each line on actual pad
void display() {
    int row, col, i;
    byte portbval=0;
    for(row=0;row<=7;++row) {
        col = buffer[row];
        if(bit_test(col,0)) output_high(COL0);
        if(bit_test(col,1)) output_high(COL1);
        if(bit_test(col,2)) output_high(COL2);
        if(bit_test(col,3)) output_high(COL3);
        if(bit_test(col,4)) output_high(COL4);
        if(bit_test(col,5)) output_high(COL5);
        if(bit_test(col,6)) output_high(COL6);
        if(bit_test(col,7)) output_high(COL7);
        portbval=(1<<row);
        port_b=portbval;  //raise clock line for this row
        port_b=0;  //lower clock line
        output_low(COL0);
        output_low(COL1);
        output_low(COL2);
        output_low(COL3);
        output_low(COL4);
```

```
        output_low(COL5);
        output_low(COL6);
        output_low(COL7);
    }
}

void update() {
    int i, j;
    long n1, n2, c1, c2;
    ++cycle;
    for(i=0;i<8;i++) {
        for(j=0;j<8;j++) {
            if(table[i][j]==0)
                bit_clear(buffer[j],i);
            else if(table[i][j]==255)
                bit_set(buffer[j],i);
            else {
                n1 = n2 = table[i][j];
                n1 *= 2; n2 *= 4;
                c1 = cycle%n1; c2 = cycle%n2;
                if(c1 < c2) {
                    bit_set(buffer[j], i);
                }
                else {
                    bit_clear(buffer[j], i);
                }
            }
        }
    }
}

int place=0;
int digit[8];
int vibrators[8];
unsigned int rate=255;
unsigned long looper=0;
char com;

void doVibrator(void) {
```

```
    int i, j;
    for(i=0;i<8;i++) {
        for(j=0;j<8;j++) {
            if (bit_test(vibrators[j],7-i))
                table[i][j]=rate;
            else
                table[i][j]=0;
        }
    }
}

//COMMAND
//interpret keyboard commands
void command() {
    int x=0;
    int y=0;
    int i,j;
    int val=0, cont;
    int cmd=' ';
    if (kbhit()) {
        cmd=getc();
        cont=1;
    }
    else {
        cont = 0;
        looper = (looper+1)%1000;
//      if(looper == 1) printf(".");
    }
    while(cont) {
        // printf("I got a %c for my input\r\n", cmd);
        if(place==0 && (cmd=='Q' || cmd=='R' || cmd=='S')) {
            place = 1;
            com = cmd;
//          printf("%c Command begun\r\n", com);
        }
        else if(place>=1) {
            if(com=='Q' && place <=5 && cmd>='0' && cmd <='9') {
                //printf("New char %c at %u\r\n", cmd, place);
                digit[place-1] = cmd - '0';
```

```
            if(place == 5) {
                x = digit[0];
                y = digit[1];
                val = digit[2]*100 + digit[3]*10 + digit[4];
                if(x>=0 && y>=0 && x<8 && y<8)
                    table[x][y] = val;
//              printf("Setting (%u, %u) to %u\r\n", x, y, val);
                place = 0;
            }
            else ++place;
        }
        else if(com=='R') {
            rate = cmd;
            if(rate==254) rate=0;
            x = rate;
            doVibrator();
//          printf("Setting rate to %u\r\n", x);
            place=0;
        }
        else if(com=='S' && place<=8) {
            vibrators[place-1] = cmd;
            if(place==8) {
                doVibrator();
                x = rate;
//              printf("Setting pad with rate to %u\r\n", x);
                place=0;
            }
            else ++place;
        }
        else place=0;
    }
    else
        // bogus data, wait for command
        place=0;
    if (kbhit()) {
        cmd=getc();
        cont=1;
    }
    else {
```

```
            cont=0;
        }
    }
}

long toggle = 0;

void vibrate() {
    int i;
    toggle = (toggle + 1)%1000;;

    for(i=0;i<2;i++) {
        if((toggle%200)<100) buffer[i] = 0x03;
        else buffer[i] = 0;
    }
    for(i=2;i<4;i++) {
        if((toggle%100)<50) buffer[i] = 0x0C;
        else buffer[i] = 0;
    }
    for(i=4;i<6;i++) {
        if(toggle<500) buffer[i] = 0x30;
        else buffer[i] = 0;
    }
    for(i=6;i<8;i++) {
        if((toggle%40)<20) buffer[i] = 0xC0;
        else buffer[i] = 0;
    }
}

//MAIN
void main () {
    initialize();
    while(1) {
        command();
        update();
        display();
    }
}
```

MR Fluid

Information about the Magneto-Rheological fluid used in the output bladder from Lord Corporation was taken from product information from Lord as well as Material Data Safety Sheets.

Product Name: Rheonetic (TM) MRX-240AS

Product Use/Class: Controllable Fluid

Description: Grey liquid with no odor. Low hazard in usual handling.

Comments specific to 240AS: Contains all non-hazardous ingredients. Water based, good non-settling behavior, limited operating temperature range. Should not be used with dynamic seals.

High Dynamic Yield Strength: Instantaneous generation of from 0 up to 90 KPa yield stress for precise real-time control.

Wide Operating Temperature Range: -40°-150°C with less than 10% variation in maximum dynamic yield stress.

Viscosity (No field): 0.08-0.25Pa-s@25°C

Millisecond Response Time: Swift response time makes the fluids ideal for use in applications requiring continuously variable, high-precision control.

Non-Abrasiveness: Proprietary additives in Rheonetic™ Fluids reduce abrasiveness, enabling devices to achieve required life expectancies.

Low Sedimentation Level: The formulation of Rheonetic™ Fluids solves sedimentation and stratification problems found in other MR fluids.

Current Density: Can energize with permanent magnets.

Specific Gravity: 3.74

Ancillary Materials: Iron/Steel

Price: roughly $500/liter

Dangers: may cause slight eye or skin irritation with prolonged exposure
No inhalation hazard.

Silicone

Information about Walco Vi-Sil Silicone V-1068 was gatehered from a Preliminary Technical Data Sheet, Material Safety Data Sheet, and catalog from Burman Industires.

General Product Description: a translucent, high strength, two component, tin catalyzed, room temperature cured silicone rubber. It is designed as a 14 Shore A, very pourable low viscosity rubber providing excellend physical properties, long library life, and accurate detail reproduction. It is easily pigmented with makes it ideal for robotic and animatronic skins, special effect skins, and props for theme parks and the film industry.

Typical properties of cured rubber (cured 7 days at 24° C and 50% RH)
Color: Translucent
Specific Gravity: 1.10
Hardness (Shore A)(4): 14
Tensile Strength, psi (N/mm^2): 450 (3.1)
Elongation (%): 500
Tear Resistance, ppi (N/mm): 115 (20.2)
Linear Shrinkage: (%): 24 Hours = 0.2, 7 days = 0.5
Coverage - cu. in/lb (cc/kg): 25.2 (909)

Teperature Range: -58° to 302° F (-50° to 150° C)

Shelf Life: 6 months from shipment.

Features:
Translucent: easily pigmented or painted
Excellent resistance to severe environmental exposure
Shore A 12-14, soft, skin-like feel; excellent photogenic qualities
High elongation and tear strength; very flexible
Low linear shrinkage
Excellent long term retention of rubber properties under dynamic use conditions
Low viscosity; easy pouring

Price: roughly $40/quart

Dangers: premixed liquids use an organometallic tin catalyst which may irritate or burn skin and eyes upon contact. If eye irritation occurs, flushing for water for at least 15 minutes should relieve discomfort.

Appendix B: Materials Suppliers

Finding good people and good products is not easy, and the following list represts the results of searching for just the right material and knowledgable contact person. If I had to do this whole project over from scratch, and could only salvage one section of my thesis, it just might be this appendix.

These people and companies were found through contacts at the lab, combing throught the Thomas Register, and talking to friends.

Magneto-Rheological Fluid (and MR elastomer):

Dr. Lynn C. Yanyo
Manager, Research and Product Development
Materials Division
Lord Corporation
110 Lord Drive
Cary, NC 27511
Tel. 919.469.2500 ext: 2329
Fax. 919.460.9648

Lynn Yanyo was very helpful in aquiring the correct MR fluid for my project.

ElectroMagnets:

Authur K. Pfeil
A.P.W. Co. Inc.
32 W. Main Street
Rockaway, NJ 07866
Tel. 201.627.0643
Fax. 201.627.6396

APW is a small company that Authur runs. He is very friendly, knowledgeable, and accessible over the phone. His windings cost about $3 each.

Medical Bladders:

Al Baris
Aero Tec Labs Inc.
45 Spear Rd Industrial Park
Ramsey, NJ 07446
Tel. 800.526.5330
Tel. 201.825.1400
Fax 201.825.1962

ATL will make custom bladders for about $150. They are "rf" sealed, and the seal never gave me problems. Al made all of my bladders, most of which had tubes on one side and I filled myself. For the last batch, I mailed the liquid off to ATL and it was sealed in place, but the bladders came back with dimples.

Silicone, Uratheyne, etc:

Donna Drexler
Burman Industries
14141 Covello Street, Suite 10-C
Vay Nuys, CA 91405
Tel. 818.782.9833
Fax. 818.782.2863

Burman Industries is a small and unique company in California that appears to supply special effects companies and hobbiests with strange materials. Everything from foam, latex, elastomers, rubber, clays, etc. to make up supplies, fake teeth, reptile eyes, and several types of fake blood. Donna is very helpful and knowledgable in finding and using materials such as silicone and uratheyne.

References

[BAE 80] R. Baecker. *Towards an Effective Characterization of Graphical Interaction.* In R. A. Guedj, P. Ten Hagen, F.Hop good, H. Tucker & D. Duce, Eds., Methodology of Interaction, 1980, p127-148.

[BAL 97] Ravin Balakrishnan and I. Scott MacKenzie, *Performance Differences in the Fingers, Wrist, and Forearm in Computer Input Control* , Human Factors in Computing Systems, CHI 1997.

[BAR 93] Sylvan Barnet, A Short Guide to Writing about Art, Fourth Edition, quote taken from preface.

[BOR 97] Ingwer Borg and Patrick Groenen, *Modern Multidimensional Scaling, Springer Series in Statistics*, 1997.

[BRA 98] Scott Brave and Andrew Dahley, *inTouch: A Medium for Haptic Interpersonal Communication*, Human Factors in Computing Systems, CHI 1997 Extended Abstracts, p 363.

[BRO 97] F. P. Brooks Jr,. M. Ouh-Young, J.J. Batter, and P.J. Kilpatric, *Project GROPE - Haptic Displays for Scientific Visualization*. Computer Graphics Annual Conference Series, SIGGRAPH 1990 Conference Proceedings.

[BUX 83] William Buxton, *Lexical and Pragmatic Considerations of Input Structures. Computer Graphics*, 17 (1), 1983, p 31-37.

[BUX 95] William Buxton, *Touch, Gesture & Marking*. Chapter 7 in R.M. Baecker, J. Grudin, W. Buxton and S. Greenberg, *Readings in Human Computer Interaction: Toward the Year 2000*, 1995.

[CAR 98] Carroll Touch, *DataSheet: Guided Wave Touch Sensors for Flat Panels*, data sheet available at http://www.carrolltouch.com/pdffiles/flatgw.pdf and related general infomation can be found from http://www.carrolltouch.com/

[FIT 54] P. M. Fitts, *The information capacity of the human motor system in controlling the amplitude of movement*, Journal of Experimental Psychology, 47, p 381-391, 1954.

[HOW 97] Robert Howe, *Introduction to Haptic Display: Tactile display*, Harvard University, http://haptic.mech.nwu.edu/intro/tactile/

[ISH 97] Hiroshi Ishii and Brygg Ullmer, *Tangible Bits: Towards Seamless Interfaces Between People, Bits and Atoms.* Human Factors in Computing Systems, CHI 1997 Conference Proceedings.

[KLO 97] Hansjörg Klock and Joachim M. Buhmann, *Multidimensional Scaling by Deterministic Annealing*, in EMMCVPR'97, Venice, accepted for oral presentation, 1997.

[KRU 90] Myron W. Krueger, *Artificial Reality*, Reading, Mass: Addison-Wesley, 1990.

[LAN 76] G. D. Langolf, D. B. Chaffin, and J. A. Foulke, *An investigation of Fitts' law using a wide range of movement amplitudes*. Journal of Motor Behavior, 8, p 113-128, 1976.

[LIP 93] James S. Lipscomb and Michael E. Pique (1993). *Analog input device physical characteristics.* SIGCHI Bulletin, 25(3), p 40-45.

[LOR 97] Product information from Lord Corporation http://www.mrfluid.com/

[LUC 95] Mark Lucente and Tinsley Galyean, *Rendering Interactive Holographic Images*, Computer Graphics Annual Conference Series, SIGGRAPH 1995 Conference Proceedings. p. 387-394.

[MCC 91] Pamela McCorduck, Aaron's Code: Meta-art, Artifical Intelligence, and the Work of Harold Cohen, 1991.

[MAE 96] Patttie Maes, Trevor Darrell, Bruce Blumberg, and Alex Pentland, *The ALIVE System: Wireless, full-body interaction with autonomous agents*, Special Issue on Multimedia and Multisensory Virtual Worlds, ACM Multimedia Systems, ACM Press, Spring 1996.

[MIN 95] Margret Minsky, *Computational Haptics: The Sandpaper System for Synthesizing Texture for a Force-Feedback Display*, Doctoral Thesis at MIT Media Lab, 1995.

[PEN 50] W. Penfield, & T. Rasmusseun, *The Cerebral Cortex of Man*. New York: Macmillan, 1950.

[PHY 98] *Physics and Media Group Purpose*, at http://physics.www.media.mit.edu/purpose.html

[PLE 98] Wendy Plesniak and Ravikanth Pappu, *Coincident Display Using Haptics and Holographic Video*, Human Factors in Computing Systems, CHI 1998 Conference Proceedings, p 335.

[PIN 97] Steven Pinker. *How The Mind Works*. W. W. Norton & Company, Inc, New York NY, 1997, 194-195.

[QUA 96] W. Quassem and M.O. Othman, *Vibration Effects of Setting Pregnant Women Subjects of Different Masses,* Journal of Biomechanics, vol. 29, 1996, pp. 493-501.

[SCH 92] Lillian F. Schwartz with Laurens R. Schwartz, *The Computer Artist's Handbook: Concepts, Techniques, and Applications*, First Edition, 1992.

[SCH 94] Chris Schmandt, *Voice Communication with Computers: Conversational Systems*. New York: Van Nostrand Reinhold, 1994.

[SIM 97] Karl Sims, *Galapagos Artistic Statement*, Appeared in ICC Concept Book, Intercommunication Center, Tokyo, 1997.

[SIN 97] Michael Sinclair, Peter Presti, Scott Robertson, *The Haptic Lens*, SIGGRAPH Sketch, SIGGRAPH 97 Visual Proceedings; also at http://www.oip.gatech.edu/imtc/html/hapticsketch.html

[SMI 98] Josh Smith, Tom White, Christopher Dodge, Joseph Paradiso, and Neil Gershenfeld, *Electric Field Sensing for Graphical Interfaces*, IEEE Computer Graphics and Applications, May/June, 1998.

[TAN 97] Hong Z. Tan and Alex Pentland, *Tactual Displays for Wearable Computing,* Proceedings of the International Symposium on Wearable Computers, 1997.

[WHI 98] Tom White and David Small, *A Poetic Interactive Garden*, Human Factors in Computing Systems, CHI 1998 Summary, p 335.

[YAR 98] Paul Yarin, John Smith, and Hiroshi Ishii, *FishFace: An Interactive Device with Integrated Sensing and Display*, submitted to UIST '98, San Franscisco, November 1998. Fishface information also available at http://www.media.mit.edu/~yarin/fishface/

[YOK 96] Y. Yokokohji, R. L. Hollis, and T. Kanade, *What you can see is what you can feel. -Development of a visual/haptic interface to virtual environment*, IEEE Virtual Reality Annual International Symposium, March 30 - April 3, 1996, pp. 46-53.

[Zel 96] Robert C. Zelenik, Kenneth P. Herndon, and John F. Hughes, *SKETCH: An Interface for Sketching 3D Scenes*, SIGGRAPH Conference Proceedings, p163-169, 1996.