The Gesture-Controller Exploration

A Study in Emotive Movement and Musical Creation

by M Bethancourt

Thesis Instructor : Marko Tandefelt

Thesis Writing Instructor : Loretta J. Wolozin

A thesis document submitted in partial fulfillment of the requirements for the degree of Master of Fine Arts in Design and Technology Parsons School of Design

May 2009

Abstract

The Gesture-Controller Exploration is a study of innovative musical instrument / controllers that investigates the relationship between movement, physical space and musical performance. The most recent incarnation, the GCe3, combines a musical software suite built in Max/Msp with an intuitive physical form to create a rich musical experience. Dipping, swinging, swaying, tilting, and turning the The Gesture-Controller sends signals to the computer running the audio software, informing its sound-making functions. This allows for a more satisfying performance, leveraging the power of the computer and helping the electronic musician to use physical means to create and manipulate digital electronic sounds in new and interesting ways.

Table of Contents

Table of Figures

Chapter I I Introduction

In the last five years there has been a resurgence and rediscovery of electronic music. Of course, it has always been there; from the keyboardist, hidden in the shadows on stage, adding subtle depth and color to the hit songs of this weeks new star; to the turntablist adding, mixing, and adapting the sounds around him. But with computing hardware getting cheaper and faster and newer, more affordable, and more professional-level software becoming available every day, an increasing number of musicians are turning to their laptops for musical performance accompaniment.

The hardware, however, has yet to rise to the challenge. Laptops were not designed to be musical instruments. They were originally conceived for work, and were specifically designed for this purpose. The Osborne I, the first portable microcomputer introduced in 1981, had a screen, spaces to insert media, and construction that folded on to a qwerty keyboard. As the development of these portable “laptops” continued, they retained many of these design features.



figure 1.1 - The Osborne I // figure 1.2 - The Gavilan SC // figure 1.3 - The IBM PC Convertible

While today’s machines are lighter, smaller, faster, and equipped with better software, the overall hardware functionality is roughly the same, namely: open computer, look at screen, and type on the keyboard. In his seminal *Digital Performance*, Steve Dixon bewails the state of PC design, calling it:

> a dreadful and pathetic interface design: an anachronistic dinosaur of a machine that places file cabinet icons borrowed from nineteenth-century offices onto a TV screen

> monitor design originated in the 1930s, above a QWERTY keyboard that, even when it was launched as a typewriter in 1878, was shown to have the worst possible letter pattern configuration. (2007)

Because of these constraints, your every day laptop “performer” becomes hard to distinguish from his white-collar cousins. Open computer, look at screen, type on keyboard. Yawn. This is not a performance. This is watching someone work.

I have been a performing musician for a number of years, and one of the most crucial lessons I have learned during my time playing live music is that an instrumentalist’s emotional investment during a performance has an impact on the sonic qualities of the music. It is possible to tell the difference between a pianist who simply taps the notes and one who truly *plays* them. Laurent de Wilde describes this phenomena perfectly when he says of the great pianist, Thelonious Monk, “He can get a sound out of anything from a Steinway or a Baldwin to a beat-up old upright. As soon as he puts his fingers on the keyboard, it *sounds*.” (de Wilde, 1997) Recently, when I began using computer accompaniment more frequently in my own live performances, I realized this inherent lack of emotional performative nature in computer-driven music. The hardware, as I mentioned above, prevents the performer from instilling emotion in the performance. It’s very nature is to be scientific, analytical, and work-oriented. It is cold and immobile and will respond no differently to a professional’s gentle touch than it will a child’s exploring forefinger.

How then can this design be improved? What if there was an external controller that could harness the power of the computer, but allows the musician to perform in a more expressive way? How would physical movement best be mapped to this sound creation? Would this lead to new and novel ways of electronic music could creation? These are the questions that launched the Gesture Controller Exploration, a study in gestural controllers for electronic musicians.

1.1 Concept

In this study of innovative musical controllers, I investigate the relationship between movement, physical space and musical performance. The most recent outcome of that study is the GCe3 (Gesture-Controller : exploration 3) which combines a robust musical software suite built in Max/Msp with a specially-designed hand-made wooden controller to create a rich and satisfying performative musical experience. The Gesture-Controller is equipped with a series of keys, force-sensors and an accelerometer that wirelessly sends signals to the computer running the audio software, informing its sound-making functions. This allows for the performer to remain untethered throughout his performance, creating a more satisfying experience for both himself and the audience. Also, by leveraging the power of the computer and introducing uncommon sound-making functionality, the Gesture-Controller allows the electronic musician to use physical means to create and manipulate digital electronic sounds in new and interesting ways.

Chapter II I Domains and Precedents

The Gesture-Controller benefits from a variety of domain studies that contextualize it in light of precedents that have been set in the past century. It expands and combines design ideas from many fields, including **electronic music**, **physical computing**, and **performance / functionality**. Through this and the following section which focuses on specific artists / projects existing in the field, I will address these three domains specifically to emerge with a more developed understanding of where my thesis project fits in relationship to its predecessors.

2.1 Domains

2.1.1 Electronic Music

Electronic musical instruments can be separated in to two distinct areas, analog and digital (fig. 1). Analog instruments directly manipulate the electronic frequencies to produce sonic frequencies, while digital instruments need to interface with a sound-producing third party.

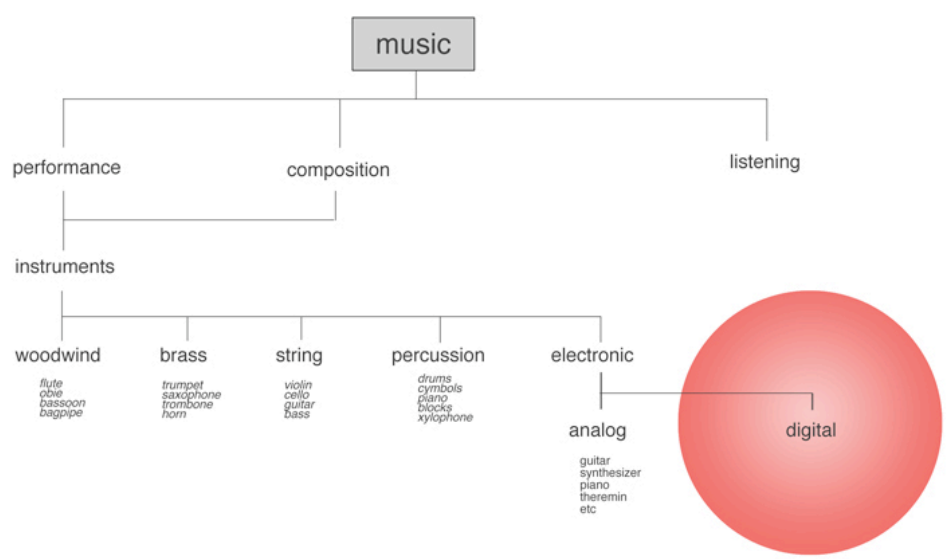

Fig. 2.1 - Digital vs. Analog music

In 1917 Lev Sergeivitch Termen (Leon Theremin) invented the theremin (Glinsky, 2000), considered by many to be the first pure analog electronic musical instrument (and one which will be discusses in greater detail below, in precedents section 2.2.1). Termen realized that by manipulating electronic frequencies the user could produce musical notes. The theories he implemented would become the cornerstone for all electronic music of the next century, most notably in the designs of Robert Moog, Leo Fender and Raymond Scott. While the theremin proved to be the foundation of electronic music, it was a difficult instrument to play, and very few people in history have been able to master it. In 1956 Scott produced the Clavivox, which involved a keyboard attached to a theremin circuit (Kettlewell, 2002). In this way, the musician could use a familiar interface to control the musical signals (see *Domain 3: Performance* for a more detailed discussion on usability issues). The designs of Moog, Fender and Scott were successful in that they made electronic music accessible to the majority of musicians who were not theremin virtuosos. However, many of these inventions sacrificed the drama of performance for accessibility. The important concept to recognize in this sub-domain is that the musician can effect and produce sounds directly on the electronic circuit.

My thesis project, however, falls in the sub-domain of *digital* electronic music (see fig. 2). In this sub-domain instruments produce no sound on their own, but require a computer to interface with the musical signal. The birth of the digital stage of electronic music can be traced back to the 1950's and 60's with Dr. Max Matthews of Bell Labs, Peter Zinovieff (founder of the Electronic Music Studios), and Dr. John Chowning of Stanford University (Kettlewell 2002). It is from these visionaries that we have analog/digital conversion, frequency modulation, and many other essentials. These leaders in the field of digital music recognized that a computer could play a crucial part in electronic music. A computer used in this way, however, can be implemented in a variety of ways. It can be used as a relay, actively effecting the circuit which produces the sounds. The computer can also make sound on its own with a physical object that sends information to the computer to control the sound it produces.

The Gesture-Controller is an example of this controller. It communicates with the computer and informs its sound-making functions.

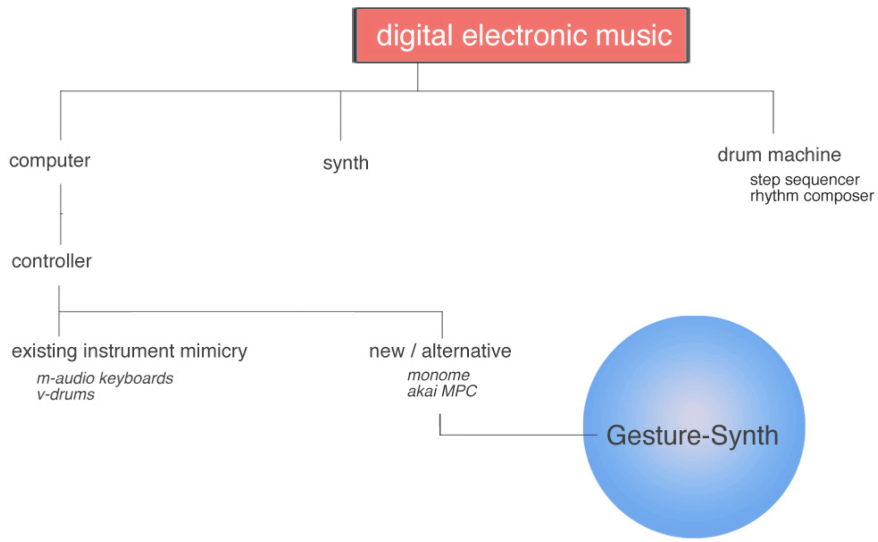

Fig. 2.2 - Digital Electronic music can be broken down into the following categories

2.1.2 Physical Computing

The greater domain of physical computing has also played an important role throughout my thesis process, and, while it may seem obvious that electronic music relies heavily on the manipulation of electronic circuits, it is an integral aspect of the project. The research and tools made available by the study of others before me has been immensely helpful both as an education in to the nature of electronics, and a launching off point for more advanced study.

Dan O'Sullivan and Tom Igoe's book *Physical Computing: Sensing and Controlling the Physical World with Computers*, published in 2004, is considered by many to be the definitive source on physical computing projects. It covers everything from the basics, such as how electricity works and description of standard components, to more advanced discussions on microcontrollers and their integration with electronic sensors. It is filled with vital information, and I referenced it continually throughout my study.

Recently, there have been numerous advances in microcontrollers, most notably with the Arduino platform. The Arduino software/hardware combination has, in the last five years, made prototyping physical computing projects extremely efficient. Developers can quickly test out code in basic circuits, see where their ideas can be improved upon, and implement those ideas immediately. The communities that have arisen around the Arduino (which help a developer to answer coding/circuit questions, provide code and sample circuits for common solutions, and give general support) are important to mention as well. These communities in combination with the Arduino's speed and ease of use have helped to make the platform a very robust prototyping option.

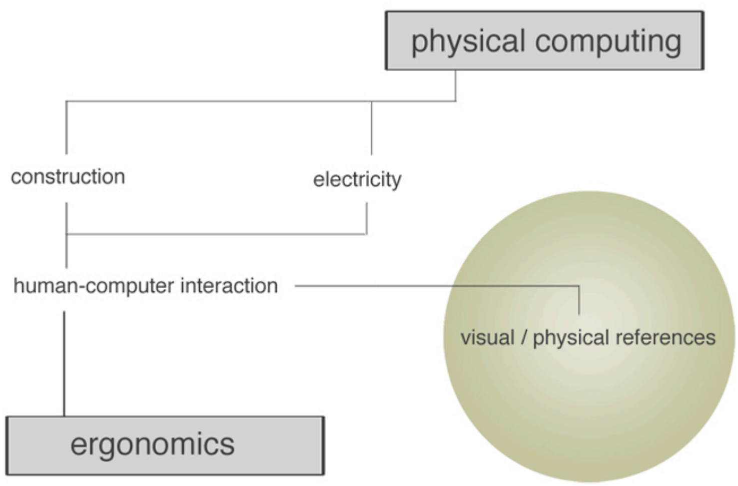

Fig. 2.3 - Physical computing in relation to ergonomics

I am also concerned with the sub-domain of electronic expectations and usability. There are certain standards that have been set in this field that are important to be aware of, regardless of end use. This is, I believe, most importantly the concept of visual/physical feedback. (see fig. 3) What does a user expect when he or she depresses a button on any electronic circuit? In addition to a manipulation of sound, most users would unknowingly anticipate a light to indicate when a function is turned on or off, or other secondary feedback, such as physical vibrations to indicate movement.

These are concepts that researchers and electronic experts have implemented for years, and was important to keep this in mind throughout the entire exploration. O'Sullivan and Igoe's book, *Physical* Computing*,* which I referenced earlier, and the Arduino forums were both great repositories of information on this subject.

2.1.3 Performance / Functionality / Ergonomics

The final domain to consider is performance / functionality. This has been an issue throughout the history of instrument building, but it became of special consideration when digital music made it possible for instruments to be whatever shape a designer desired. When the physical object is no longer responsible for the production of sound, these designers can focus more specifically on novel ergonomics.

It is interesting to see how this notion has been applied throughout the evolution of electronic music. From the beginning, it was an issue. In the 1960s Bob Moog and Don Buchla took very different approaches to their synthesizers. Moog decided that a

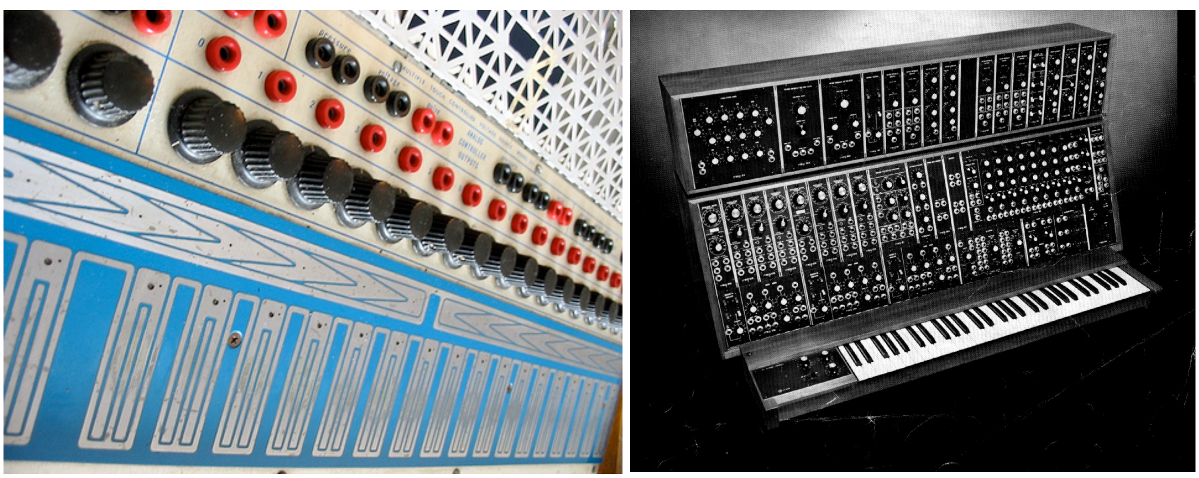

Fig. 2.4 - Buchla's 200 series keypad (left) and Moog's Moog Modular system with keyboard (right)

keyboard interface would be the easiest way for musicians to migrate over to the electronic music world, and, seeing that his modular Moog systems became the first widely used synthesizer, he was clearly correct. Buchla, on the other hand, thought that electronic music should not be tied too closely to existing use scenarios, and many serious performers find his 200 series Music Box system of touch-pads superior. Many, however, had difficulty learning a new technique, and because of this his solution was never widely implemented.

In recent years, however, alternative music controllers have started to crop up with more frequency. Starting in the 1970s, when east coast DJs started reappropriating their turntables for alternative purposes, popular musical performance has been progressing to a new level. As musicians have become more accustomed to computer-driven music, the systems that they use have become less and less focused around the musical keyboard. This can be seen in the rise of the Akai Music Production Center (MPC) and other sequencers. These systems use a grid of buttons to control pre-recorded sounds and musical loops. The system of gridded buttons is also becoming a standard, with alternative controllers such as the Monome (See section 2.2.2) and even M-Audio's Oxygen line of keyboards adopting the technology.

For the Gestue-Controller exploration I desired to examine new interfaces for the physical controller, but it was important to not be too confusing. Buchla's system may have been more interesting, but many musicians, finding it too difficult to learn, decided to stay with the Moog. In my designs, I sought to retain the innovation of Buchla and the usability concerns of Moog. Regarding innovation, the gestural element of the controller was paramount. I feel that it explores a possible solution to a means of musical expression that has been missing from electronic music. However, I realized from my research that this innovation would be overlooked and unsuccessful if it was not married with strong usability prototyping. Moog's success taught me the importance of repeatedly testing and re-testing my interface ideas in order to be certain that they would be comfortable, natural, and enjoyable for my user base.

2.2 Precedents

2.2.1 The Theremin

In 1919, Léon Theremin (born Lev Sergeyevich Termen) invented his namesake instrument, the theremin. It single-handedly started the electronic music revolution, and without it we would certainly have no Bob Moog, for it was the theremin that kindled his love for synthetic music (Pinch and Trocco, 2004). It is also famous for being the first instrument that the performer did not not need to touch to play. It forced people to rethink the way music is performed, and has contributed to numerous pop songs over the years. Perhaps most notably is its prominent place in the Beach Boys' 1966 masterpiece, *Pet Sounds* (Fusilli, 2005). I am interested in the ideas that Theremin first explored, namely that of controlling sound created by electronic means with physical gestures. What is especially fascinating about Theremin is that, even though the instrument used electronic impulses, there was still an inherently performative nature to the instrument.

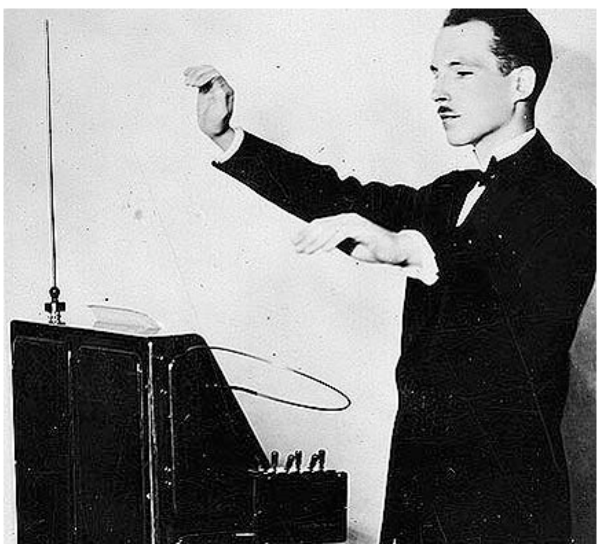

Fig. 2.5 - Theremin playing his namesake creation // Fig 2.6 - Robert Moog's Theremin Model 201

2.2.2 Monome / Tenori-on

Brian Crabtree's open source Monome device is quickly becoming the Theremin of today. Widely adopted, almost-unanimously praised, and well-loved, is has found its place in the popular production of music. Many people have latched on to its adaptable interface and have started making music in ways they had previously only thought possible in their heads. Japenese artist Toshio Iwai's Tenori-On, released soon after Monome's original 40h, is similar in that it is a gridded interface of tangible buttons but is far more robust, sporting two speakers, an lcd screen. It is, however, not open source but has a number of built-in functions. In my explorations I desired to combine the open-source-esthetic of the monome, allowing for users to have access to the plans, code, and protocol for the final product, with a rich software suite for music creation, like the Tenori-on. This will allow novice computer programmers to pick up the device and “plug-n-play,” while also providing more advanced users with a stepping off point from which to make improvements, explore/expand on new functionalities, and perhaps improve the experience for later users.

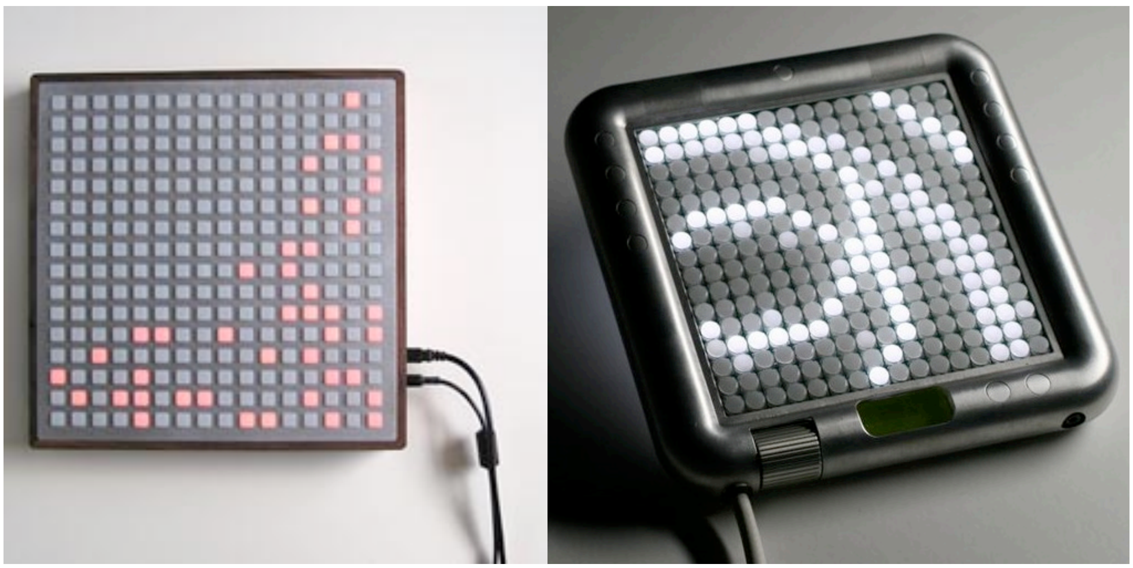

Fig. 2.7 - Monome 256 (left) and Tenori-On (right)

2.2.3 Thingamagoop / Thingamakit

Bleep Labs, of Austin, Texas, have released two interesting music machines in the last few years. The first is the Thingamagoop, which people are drawn to partly because of its personality. When you see the weird mouth and googly eyes, it is hard not to feel connected with this musical device. It could easily be used in a performance, much like a Theremin. The squelching noise may be too much for some listeners, but music-makers across the country have begun buying these up for adding texture and depth to performances. In 2008 Bleep Labs announced that they would be selling kits of their Thingamagoop. They called this customizable Thingamagoop the Thingamakit, and soon they had a flickr page full of new versions of their original idea. This community is similar to the one that Monome has created, and it is this idea of community that I think is important. These communities push the design ideas, create new modifications on the hardware/software, and help bring newcomers with questions they may have. During my exploration I kept in mind the notion of this community.

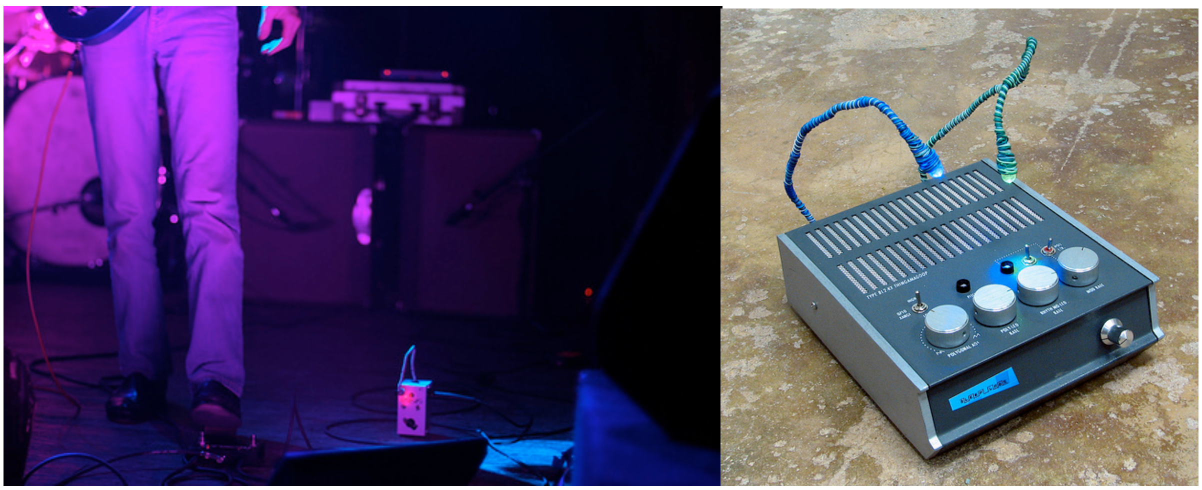

Fig. 2.8 - Thingamagoop on stage with Peel (left) and Dr. Bleep's Thingamakit prototype (right)

Conclusions

Since the arrival of electronic music, there has been an inherent tension between the sound quality an instrument is able to produce, the ease of playing, and the performance aspect. The ideal solution would merge all three of the domains I have described above to produce a controller that builds on all three of these domains. The goal of the Gesture-Controller, then, is to create a music controller that could make interesting sounds, leverage the power of a computer, allow for a satisfying performance experience, and leave room for improvements in the future.

Chapter III I Development

This chapter examines the different stages of the Gesture Control Exploration, from initial concepts and use scenarios to the final iteration presented in the spring semester. I will begin by describing the final controller and software solutions in my exploration, and then discuss the conditions and concerns that have caused the exploration to evolve over time, giving a full description of the different design solutions and experiments.

Fig. 3.1 and 3.2 - the final implementation of the Gesture Controller Exploration, the GCe3

3.1 GCe3

The GCe3, the most recent controller in the Gesture Controller Exploration, uses natural human movement and well thought out sensor-to-sound mapping to create a more performative tool for the electronic musician. It is made up of two integral systems, the actual GCe3 hardware, and a software suite of musical instruments that show off the capabilities of the controller.

3.1.1 GCe3 Hardware

The first thing that the user may notice about the GCe3 is its construction. The overall shape of the controller is like a flattened sphere, more wide than tall. It is made out of solid wood, lightly varnished for durability. The bottom of the controller has eight brass keys, modeled after trumpet valves. These keys, when pressed, trigger small tactile switches hidden within the controller, and custom-built springs provide a nice range of movement and resistance. These keys are the basic triggering mechanism of the controller. With them, the user will play notes and trigger sound events.

Also hidden inside the controller are a three axis ADXL202 accelerometer and an Xbee Series 2 wireless transceiver. The accelerometer read the tilt of the device (forward, back, left, and right) on the horizontal plane. This information is sent to the computer via the transceiver. These two components in particular help to create the main gestural atmosphere of the controller, allowing the performer to intelligently, artistically, and meaningfully move the controller through space unencumbered by the powerful computer program outputting the music.

There are also two leather pads where the user's palms will be while holding the instrument. At first these may seem simply decorative or for general comfort, but in reality there is a small force-sensing resistor (FSR) and some padding hidden beneath each of the two leather pads. These will provide the user with a way to incrementally (and, of course, gesturally) modify certain sounds and programs in the GCe3 software.

Lastly, there is a three-way navigation switch protruding from the top of the device, about here the user's right thumb would be. This is for switching between the various program in the software suite and provides a more robust music-making experience. With this functionality in place, the user can begin in one program, playing a certain type of sound, and then switch to other programs, overlaying sounds, applying effects, and creating a deep tapestry of music.

3.1.2 GCe3 Software

The software suite created for the GCe3 uses the incoming data from the controller, and parses it to crate the most natural musical performance possible. The program, built in Max/Msp, has six main modes. These are basic controls, bpm, freeSample, polySynth, randomizer, and the amplifier. The user presses the directional switch on the top of the controller forward or backward to scroll through the different programs, and each one is highlighted in a band of red to indicate which program is being controlled. (For example, in figure 3.3 the bpm function is currently selected).

In the basic control mode the user can choose which version of the controller they would like to use, update their baud rate, and turn the device on and off. This basic section provides simple overall controlling of the software so that the user will never have to touch their computer.

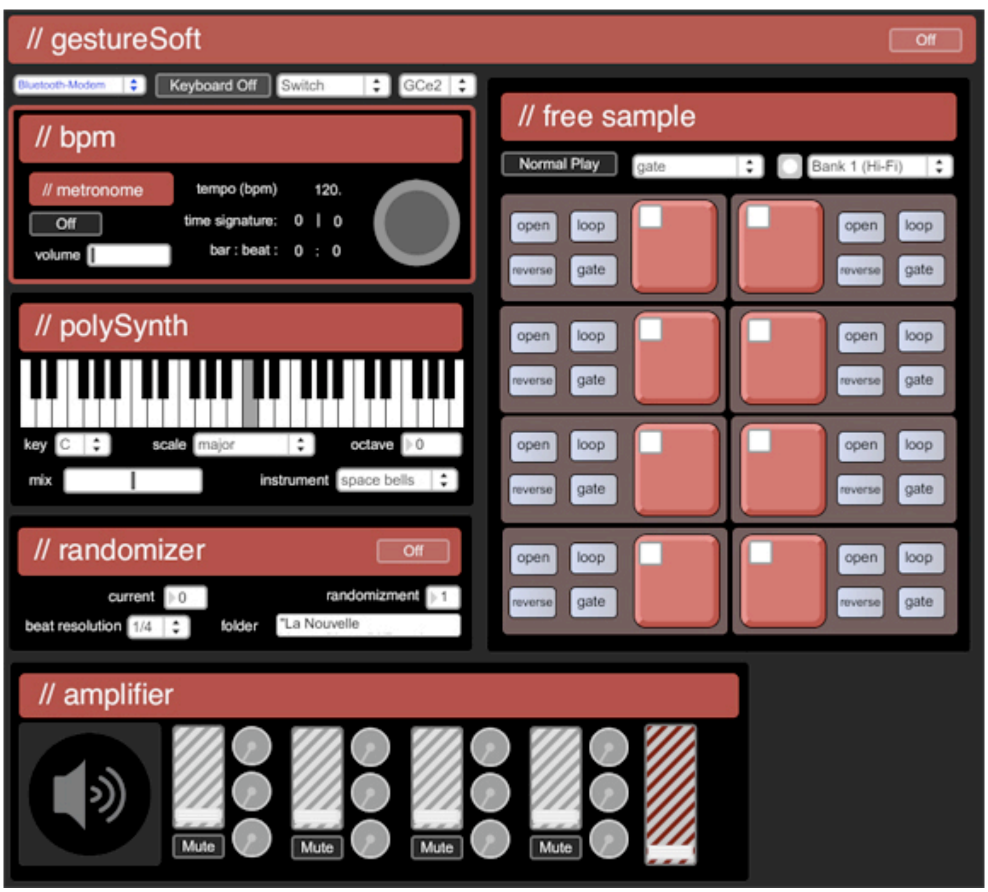

Fig. 3.3 - The gestureSoft Max/Msp software suite

The second function is bpm. In this program the user can turn on the metronome by pressing Key 1, tap Key 5 repeatedly to create a tap tempo, or use the left and right tilt to incrementally modify the overall tempo of the software.

freeSample, the third program in the software suite, is a simple sampler with some interesting performative capability. Each of the eight keys triggers a pre-loaded sample, and double-clicking on that key starts the sample with the jogging feature activated. When jogging is activated the user can tilt the controller left and right to speed up and slow down the sample, almost like scratching a record. The pressure pads activate the sample muter. The user presses the pads and then chooses which samples he wants to mute. Once he presses those samples, they will cut out until either the buttons or the pads themselves have been released.

polySynth is a powerful polyphonic synthesizer with numerous pre-loaded voices. The eight keys are the eight notes in an octave of any given key. The user can choose between most widely used keys and scales. Tilting the controller left and right transposes the octaves being played up and down the "keyboard." Tilting the controller backwards turns on a chorus effect, and squeezing the pressure pads activates a flange on the notes being played. In combination, this system provides a rich synthesizer that the user can truly perform with.

The randomizer program is a random sample generator. Prior to the performance, a user can drag a folder of samples (drums, animal sounds, notes) into the window and it will populate the program with all of the provided sounds. It is automatically tied to the overall bpm of the system, so when the user presses Key 1 the randomizer begins to play these random sounds at the specified interval (quarter notes). This interval can be changed (eighth notes, 16th notes, 32nd notes) by pressing Button 5. Also pressing and holding Key 6 while tilting the device left or right changes the overall frequency of the samples, while pressing and holding Key 2 while tilting the device left or right decreases and increases (respectively) the number of samples the program chooses from.

The final application provided in the gestureSoft suite is the amplifier. With this the user has complete control over the volume and effects of each of the previous programs (freeSample, polySynth, bpm, and randomizer). Keys 1 - 4 are the sound programs, and Keys 5 - 8 are for the parameter to control (volume and effects 1 - 3). So, if a user wanted to turn up the volume for the bpm program, they would simple hold down Keys 3 and 8 and tilt the controller to the right. This way the user can make these changes in sound without needing to return their eyes back to their computer,

3.2 Process / Experimentation

In order to develop a truly useful controller, I undertook many experiments and iterations to test design ideas and theories, Many experiments, trials, failures, sketches, and user tests have led to where the Gesture Controller Exploration is today. The size, shape, functionality, material, sonic qualities, and many other properties were narrowed down and tested repeatedly with these explorations. This section dissects these experiments and highlights what was learned from each exploration and later applied to the final implementation.

3.2.1 The SynthMonster - Initial Exploration

3.2.1.1 Implementation

In the spring of 2008, I began my first explorations with non-traditional interfaces. I felt that, in general, most musical interfaces of the modern era were angular, cold, and of an unforgiving space race style, so I was interested in seeing what could be done with curves, soft lines, and warmth. I decided to explore how users would interact with a feature-less ball of fur that made music based on how you held it. I tried several different furs, and my tests showed that a standard teddy bear fur was the most liked. I also experimented with different sizes of the device, from roughly the size of a toddlers fist, all the way up to a small soccer ball. In the end, I chose to go with a softball-sized shape. It got the best response out of my tests, and would be easy to handle.

The SynthMonster, as I began calling this version of the exploration, used a gyroscope to detect its x and y tilt. This information was processed by a PIC chip that created the music. I decided to have the x-tilt control the speed of the sound being created, and the y-tilt affect the pitch. I used an alternating pattern between c5 (middle C) and a random note in 3/4 time signature. With this implementation the user could discern the pitch change in the c5 note, causing a much more responsive impulse. My hope was to create an easily comprehensible sound pattern, causing the user to interact with it more as a musical instrument than an object of curiosity.

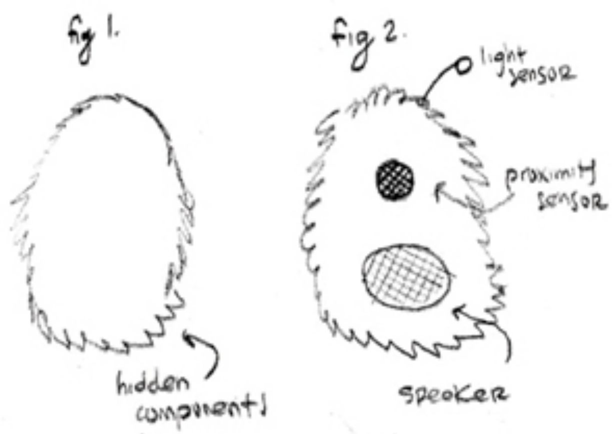

Fig. 3.4 - Early designs for the SynthMonster // Fig. 3.5 - Final SynthMonster implementation

3.2.1.2 Evaluation

Overall, the exploration was a modest success. The final user tests proved that the interaction was entertaining and, more importantly, it helped to show in which future directions I would like to go. First, I would switch to the Arduino micro-controller for the rest of the exploration. I saw that a programming environment that would provide a means to quicker prototyping and idea implementation would be better suited for these experiments.

Secondly, I realized that I needed a more thoughtfully mapped out sound-to-sensor data flow. Randomly generated noise was entertaining, but a more mature and subtle orchestration would truly engage users. Along with this, I knew that there needed to be more control over the sound than just the tilting of the object.

Also, I decided to turn my focus on controlling external, computer generated sounds rather than internal noises. The sounds capabilities from a single chip proved rather limiting, and, while multiple chips could create more complex sounds, the space constraints proved too limiting for the complexity I desired. I decided to use the controller to manipulate sound created from Max/Msp. This allows for much more complex sound creation and easily implemented iterations as I proceed, and, if I eventually decided to make the controller a stand-alone instrument, the Max/Msp synthesis could be ported to a single chip with the help of an electrical engineer.

3.2.2 Button Exploration 1

3.2.2.1 Implementation

The second phase of experiments began in the fall of 2008. First, I built a Max/Msp patch that allowed the user to play musical leads and/or patterns. The patch used simple equations to generate every key in standard western notation (A through G, including all of the sharp/flat combos - i.e. A#/Bb) and applied the four most common scales: major, minor, harmonic minor, and blues. I was curious if this would be something that the user would find interesting. Would they find the option to easily play melodic patterns a benefit, or would this notion, already so prevalent in existing instruments, seem tired and not useful? To answer these questions, I attached the series of eight note buttons to a block of wood and handed the prototype to the participants without any prompting to see what a user might do without any prior knowledge.

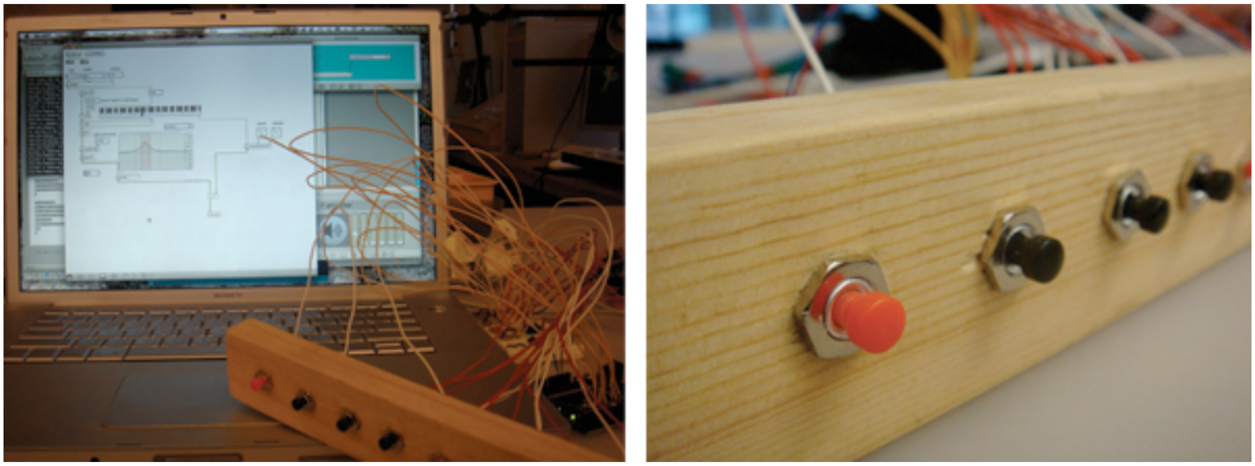

Fig. 3.6 - Button exploration

3.2.2.2 Evaluation

My users generally found this fun and interesting. Most participants asked for access to changing the sound. Some, having seen my earlier prototypes for the gyroscope-influenced synthesizer, tilted the block of wood, hoping to effect the sound. This strengthened the idea to me that my original gestural construct was of interest and natural for the user. As for aesthetics, I used black and red buttons and chose a purely decorative pattern of red, three black, three red, and black. At first I considered alternating black and red, but I didn't want to make the buttons look too much like a piano keyboard. Overall the users found the choice of colors confusing, so I decided to implement a single color button solution in the future.

Overall, many of my questions were answered in these experiments. Users liked the idea of note-based performance, and they found it useful over semi-structured gestural performance. Also, they desired some advanced gesture-based manipulation on top of the structured note-making. I realized that, with this groundwork, I could apply different gestural augmentation that would add to the experience, rather than distract and confuse from the start.

3.2.3 Ergonomics Exploration 1

The first ergonomic experiment focused on the size, shape, and general button-placement for the device. After completing earlier experiments, I realized that it was crucial to develop the shape and size of the instrument before moving on to other factors. I saw that, only after the physical properties start to take shape, could the sonic properties be adequately mapped. In other words, the form would beget the function. To answer the questions I set up 3 small experiments, each focusing on a specific aspect: shape, size and button-placement.

3.2.3.1 Shape User Test

The first question I needed to answer was one of shape. My initial designs were for a spherical instrument. Most instruments have a set front, back, top and bottom. It is clear when you approach an instrument with a keyboard that it will react like a piano does. Few people with any prior musical experience will pick up the piano and try shaking it to see what happens. Because of this, I believed that an instrument without any front, back, left, or right side would bring more discovery and interest to the user. I thought this would alleviate the preconceived ideas of how the instrument might work. I also desired to create an instrument that was not directionally oriented for my own selfish curiosity. Even after abandoning the fur-covered housing from the earlier iterations, I still believed that this was the most interesting solution. However, I needed to test this hypothesis with some actual users.

To test this hypothesis I set up a simple experiment. I had three shapes that I asked participants to look at, and I asked them which they would prefer. These were a) a sphere, b) a cube, and c) a half-sphere (like a globe cut in half). I told each participant not to worry about the size and only focus on shape, and I asked him/her whether they believed this to be an interesting shape for a musical instrument. This experiment took place on Thursday, September 25 in the 10th floor lab in the CDT department of Parsons, the New School for Design. I interviewed 15 participants. There were slightly more females than males, and the age ranges were from 25 to 32. A few of the participants were wary of the sphere. Lynn, a 27 year-old female, expressed concern over how the sphere would rest. She was afraid that it would roll away during a performance, creating unwanted sound. Most of the participants liked the idea of the half-sphere. Erik, a 29 year-old male, liked that this would make the device easier to store than a sphere. None of the participants desired to see a cube-shaped instrument.

3.2.3.2 Size User Test

Next came the issue of size. Would users enjoy using a smaller, easily wielded, instrument, or would they rather have something with more substance and weight? While it is true that different instruments can easily span this divide, I was convinced that, for my purpose, a good middle ground was the best solution. I wanted to prove that something not substantial yet something that felt solid in your hands but could easily fit in to a bag, would be optimum.

To answer this design question I built three identically shaped objects using the previously tested half-sphere shape and, over a week-long period, asked users which they preferred. I made sure that most of the participants were asked independently from each other so as not to sway their opinions. These sample objects were made out of three round aluminum mixing bowls purchased at a restaurant supply store on the Bowery. Their diameters were 6.5 inches, 8 inches, and 9.75 inches. The larger mixing bowls also were taller, and their heights, in order of diameter, were 2.25 inches, 2.75 inches, and 3.8 inches. I cut particle board to fit snugly on the open end of each of these bowls and secured them snugly with screws. The particle board added to the height of each mock-up by a half an inch. I believed adding the particle board cover to the object would make it look less like a utensil for cooking. This was achieved. Many of my testers asked where I got the interesting base for the mock-up, not able to tell at all that it would be used in the kitchen.

Fig. 3.7 - Size exploration

During the week of October 1 to October 8, I interviewed 10 people in the CDT department at Parsons, the New School for Design. They ranged in age from 23 to 38 years-old. Interestingly enough, they were unanimous about the size of the instrument. Every single person liked the smaller version of the instrument the best. I had thought that the medium sized mock-up might have a bit more authenticity to it, but this was not the case. The entire test group preferred the smallest mock-up. Akiko, a 23 year-old female student from Japan, thought that the smallest version would be easier to use and would be "really easy to throw in a bag." Alia, a female 23 year-old student from Ontario, Canada, completely independently, said almost the same thing. Without prompting, she actually put the device in her bag to see how it would travel. This was not something that I had even considered up to this point, being so focused on the musical properties. Both of these testers, in particular, would use controllers such as this for visual art-works, so this was very interesting to discover. It was clear from this experiment that portability was just as much a factor to success as ergonomics.

3.2.3.3 Button Placement

The last design question I looked at was button placement. Specifically, I was curious how the user would prefer the orientation of the device. Would the buttons be better suited on the flat, wooden side of the device or on the rounded convex side? Would the user like to hold the flat side away or towards himself/herself?

To answer this question I took the most up-to-date prototype (the smallest half-sphere bowl with attached wooden top) and drew buttons on both the wooden side and the convex bowl side. The "buttons" were simple areas drawn in pencil and marker. This experiment also took place in the 10th floor lab on Monday, October 6, and Monday, October 13. I asked 4 of the people who participated in the second experiment to test these two options. The first week (October 6) the buttons were drawn on the wooden side of the device. In this version, each user still held the device with the wooden side facing himself/herself. When prompted to hold the device the other way, most felt that this was awkward and unnatural. The second week (October 13) the buttons were erased from the wooden side and drawn in marker on the convex side. The same four users were asked to test this modification. They unanimously agreed that this was more natural. Catherine, a 30 year-old female said, "oh, yes, this feels great. The other was way too awkward."

Fig. 3.8 - A user testing the ergonomic button placement exploration

3.2.3.4 Evaluation

The users seemed interested with controllers in new and novel shapes. However, I learned that there needed to be some kind of discernible orientation. I had not considered the difficulty or confusion that would arise from an instrument with no obvious "front". A spherical shape with no button would be too bizarre for the first-time user to comprehend. While repeated uses might make this more clear, it was clearly important to have at least some aspect of the device easily understood at first glance since the goal is to have the user naturally desire repeated encounters with the device. I also wanted the device to be fun to use, not confusing and uninteresting. The sphere proved to be too ambiguous. Also, the users needed some way to put the device down when not using it, and the sphere had the danger of rolling away. It was easy to solve this dilemma with a flattened bottom. The best choice, from user testing and further thinking, was clearly the half sphere shape. It has the novelty of the sphere while keeping in mind the usability concerns and preconceived notions of most users.

Also, users want a smaller sized device. It was important to have something hand held, especially for newer electronic music performances. Users would be unwilling to abandon their keyboard setup, and a device that can supplement their current setup would be more widely accepted. Portability was also necessary. Users want something that can easily be stored in a bag. Apart from the experiments, we can look at the success of Monome's 40h as precedent. While they have come out with larger, perhaps more robust designs, the smaller portable 40h is still more widely used.

Finally, the primary buttons would be placed on the bottom of the device. Every user fund this most interesting and useful. Most agreed that sliders or touch sensors on the top could add to the robustness of the device, but it was the first priority to have a set of buttons easily accessed while holding the half-sphere with the convex side facing away from the user. This makes sense in relation to the human hand. It is understandable that a user would like to hold something in the most natural way, and this is clearly the way they want to hold the instrument.

3.2.4 GCe1

3.2.4.1 Implementation

During all of the ergonomic experimentation I was simultaneously building an early prototype to test functionalities with my software. This exploration did not take in to consideration all of the things I was learning about size, shape, etc, but instead focused on how the various buttons and sensors would best be used to create music.

I built this prototype in to the medium bowl I had used in my earlier user tests. The wooden discs remained on the top. I installed the same simple momentary switches from my previous test as well, but added them along the bottom of the device, in a hand shaped position I felt would be easier to trigger. In this prototype I also installed an accelerometer, and it, along with the eight buttons informed the music making software I was beginning to build in Max/Msp. These early tests and software construction, would become the basis for the polySynth program in my final exploration (see section 3.1.2). The eight buttons triggered the eight notes in an octave and tilting the controller left or right decreased and increased the pitch, respectively.

3.2.4.2 Evaluation

I took two main understandings away from this, the first real prototype of the device. First, I realized that the sensors I had installed were not gestural enough. Titling left to right and even front to back was not in itself interesting. A joystick could do similar movement. I saw that what would make it truly gestural would be a combination of all of these components, along with new sensors that I knew had to be added, in a way that would bring music to life out of the complex and varied movements that a user could make.

Secondly, I was further reminded of the importance of ergonomics. For the device to be truly gestural, it needed to become fluid in the hands of its users. This did not mean there would be no learning curve. Much like a guitar sits on the knee and invites the hands to play (but still takes some practice to become accustomed to) so my new instrument would have to take in consideration the various shapes of hands and most comfortable holding positions. Only in this was could it be accomplish the goal I had in mind.

3.2.5 Ergonomic Exploration 2

3.2.5.1 Implementation

From this knowledge, I set out on a second round of ergonomic research, to improve the way in which the instrument felt in the users' hands. I used clay molds to see what finger placement was the most natural for a group of users. I asked a series of participants to take a variety of clay shapes (ovals, cubes, long slender “snakes,” etc.) and mold them as they saw fit. I asked them to hold them as comfortably as possible and squeeze their fingers. Using the results of this experiment, I was able to map where the most comfortable positions for the majority of users could be. By testing users with smaller and larger hands, a medium ground could be reached.

Fig. 3.9 - Three results from the second ergonomic exploration

3.2.5.1 Evaluation

This exploration primarily showed what I had already thought to be true. Two arcs of finger placement points, wider at the forefingers and narrower between the pinky fingers, was the most common. I saw too that there were grooves where the fingers lay upon the clay. I realized it was not enough for the fingertips to have obvious pressure points, the entire fingers had to be taken in to consideration. Also, users tended to hold the shapes so that they were level to the ground. This information became invaluable as I proceeded on my construction for the second Gestural Controller Exploration (GCe2).

3.2.6 GCe2

3.2.6.1 Implementation

After the initial construction, testing, and software development of the GCe1, I set out to build an exquisitely designed version of the controller. All of the findings from the previous iterations were considered, and used to expand on the functionality, beauty, and utility. One thing that really set this version of the controller apart from the earlier explorations was the material used. I decided to build a wooden case for the controller, as I thought this would best convey the elegance of instruments of the past. Two wooden bowls were sanded to fit flush on each other, and then the whole shape was further sanded to be a cohesive whole, with sections especially shaped to create obvious places for the user's hands. The final object was then sanded, buffed, and stained for a smooth and refined appearance

Many of the earlier electronic components of the controller were the same. It still had the eight buttons along the bottom as sound triggers. However, these were embedded in a more ergonomic way. Momentary push-buttons were hidden inside the instrument. As the user pushed round wooden buttons into the controller, they would trigger the buttons hidden within. This endowed the whole controller a more organic essence. The second version of the controller also included the accelerometer, though I realized that I would need to develop a more robust sound-creation function to incorporate this functionality more appropriately. Force sensors, embedded in soft leather swatches, were also added where the user's palms met the physical shape. This added an even more robust and interesting way to trigger or manipulate sounds, as the user could squeeze the instrument in their music creation.

Fig. 3.10 - The Gesture-Controller Exploration 2 (GCe2)

3.2.6.2 Evaluation

In the winter of 2009 I was able to exhibit this version of the controller at the Handmade Music Night put on by Create Digital Music and Etsy. Around 75 people tested my controller and I received some amazing feedback about its successes and failures.

The single biggest question that I received about the device was, “what else does it do.” I realized that there was interest in the controller, but that my simple synthesizer functionality was not enough. At least to show what other capabilities the controller could have for people, not all of whom desired a lead or pad instrument, I would need to design various instrument modes. This led to the multiple programs in my final software and the method of switching between them all with the three-way switch.

I also realized what a hinderance the usb cable was. While providing a very powerful and fast connection between the device and the computer, it was hard for users to get a truly gestural motion if they couldn’t spin around, place it on the ground, spin it, or do whatever else might come in to ones mind in a performance setting. This caused me to develop the wireless functionality that is implemented in the final version.

3.3 Conclusion

Each exploration that took place during the last year of study helped to shape and influence the final implementation of the controller. I learned a great deal about basic ergonomics, electronic sound creation, music software functionality, and sensor-driven physical computing solutions, among a myriad of other design considerations. These experiments helped to narrow down the possibilities and focus the design questions more specifically toward a better solution. In the next chapter I will evaluate this final implementation, examine its flaws, and, looking toward the future, describe what steps will be most beneficial for further development.

Chapter IV / Evaluation

The GCe3 is the culmination of this phase of the Gesture-Controller Exploration. As I hope to continue this research in the future, it is important to reflect on what I have accomplished and learned thus far. In this next section I will analyze the hardware and the software portions of the project and examine how successful they were, both individually and in their union with each other. This evaluation will help inform the future of the exploration, and I will briefly discuss the various changes that I desire to make and the possible directions that the exploration could take as I continue with thie research in the future.

4.1 Analysis

4.1.1 Hardware

Major advancements and achievements were made throughout this research and exploration process. The user tests, experiments, and various trials and errors helped to hone and improve the physical form of the controller. The addition of wireless technology was absolutely essential in realizing the true gestural capabilities of the project. Tests with the newly wireless controller revealed it was indeed easier to use, creating a much more fluid and expressive experience. The overall size and shape were also well received. Users found it easier to hold and much more comfortable in general. The “valve-influenced” buttons made it possible for users with different sized hands to use the controller more easily.

Overall, the controller, with the changes made throughout the years experiments, was a successful final implementation of the original concept. It created the experience that I desired, allowing the user to separate himself from the computer while still having access to the powerful music making capabilities. The size, shape, and functionality was successfully refined to provide a comfortable interaction.

4.1.2 Software

The gestureSoft software suite that I developed throughout the last year proved to be successful in a performance setting. While some earlier programs proved to not be as useful and were abandoned along the way, I was able to create and refine a handful of very powerful and expressive programs during the course of my exploration. The sampler, synthesizer, amplifier, and other various instrument functions were easily combined to solely create a musical performance.

The gestural nature of the controller seems successful in this final prototype. The user can easily and effectively control and influence the sounds being created, and the link between software and hardware is understandable and effective. There is, as to be expected, a learning curve for some of the programs, but easier synthesizer programs can be grasped in a matter of minutes. The modularity of the software to hardware connection still needs to be fleshed out more thoroughly and thoughtfully. If I truly desire the end user to build his own programs for the controller, I will need to come up with a better main software/hardware protocol program, perhaps in Open Sound Control. I can look at Monome's monomeSerial as a precedent for this future development. In general, however, I think that the clear usability and enhancement to live performance that the controller exhibits shows its success and is a viable avenue for future development.

4.2 Future Directions

As I continue to develop the ideas and explorations, there are a few different directions and implementations that I would like to attempt. First, I am still interested in a smaller version of the controller that could be held in one hand. It would be interesting to see how this could be used to effectively influence musical performance. I can also envision this as a collaborative and modular system, where a user could have any number of smaller controllers that control various aspects of a performance and interact both with the computer and with each other. I feel this could be a really interesting path to explore, especially if I am able to develop a software application that makes the programming of the modular controllers fluid and easy. This could allow the user to have a great deal of control over their controller configuration and, thus, possibly provide for a more specialized experience.

Secondly, I would like to continue exploring the visual feedback implementation. I realized in all of my user testing that this was lacking in my final design. The screen provides some useful information, but the nature of the device is to separate from the computer as much as possible. For this to really happen there needs to be a way to see what effects, program, and sounds you are controlling from the device itself. I can see two possible methods for this implementation: an installed screen or lights. Of these two solutions, I think that the screen would be the most effective. My only concern is that it may mar the overall aesthetic of the device. For this not to be the case, I know that there is a lot of prototyping that would need to happen.

Third, I would like to continue improving the tactile feedback of the controller itself. The squeeze function is not where I would like to see it. I think that there could be an even more satisfying and deeper squeezing mechanic where the hands travel farther and with a nice resistance. The buttons are definitely a step in the right direction, but I still think they can be improved on as well. I would like to see more movement and a more satisfying "click." Also I would like to implement a vibration feedback for every button press or as the user turns the device, which I feel would give the whole experience a more whole impression of spatial/positional complexity.

Lastly, I would like to experiment with new sensors and sound to sensor relationships. These do not have to be for a finished product, but I am interested in a continued exploration in gestural control for computer music. Are there even more interesting and satisfying movements? What about a controller that you can hit or shake? How about a controller that reacts to light? Or its own sound output? What benefit could an external microphone lend to the experience? I believe that these small experiments could lead to some very interesting results for future implementations.

Chapter V I Conclusion

I started out this year unsatisfied with the current state of computer music controllers. I desired to explore and develop a possible solution to the lack of performance-based interaction in laptop performances. This curiosity led me to experiment with various sensors, Max/Msp patches, sound sets, and, finally, combined hardware and software solutions that create a unity between an electronic music performer's motions and his music. Throughout the course of the exploration I learned what was successful, what needed to be developed at a later date, and what needed to be abandoned all together. I believe that I answered this underlying problem posed thesis, namely that there is indeed a need for these gestural controllers in the world of laptop performance. They are viable and, in the future, should be developed more thoroughly to expand the artistry and desired visual interest of computer-aided musical performances. While my final prototype is not the electronic computer-controlling instrument that will set the music world on it's ear, it is certainly proof that these ideas are worth considering. I believe that further development and research into the usability, interaction, performance, functionality, shape, construction, and sonic properties of these computer music controllers by myself and others may lead to a more meaningful, powerful, and enjoyable performance experience both for the laptop musician and the audience. I have explored a few possible ways that a musician could use movement to influence performance, but there is clearly an opportunity for further study. The computer is a powerful tool to create exquisitely beautiful and thoroughly interesting music, and further development of a truly performative way to control these sounds with an exquisitely crafted tool that perfectly connects both sweeping and delicate motions to the creation of sound will enrich the experience of the audience and performers alike.

Works Consulted

Books

Attali, Jacques. (2003). *Noise: The Political Economy of Music*. Translated by Brian Massumi. Minneapolis: University of Minnesota.

Auslander, Phillip. (2008). *Liveness: Performance in a Mediatized Culture*. London: Routledge.

Collins, Nicolas. (2006). *Handmade Electronic Music: The Art of Hardware Hacking*. New York: Routlage.

Cox, Cristoph and Daniel Warner. (2004). *Audio Culture: Readings in Modern Music*. New York: Continuum.

de Wilde, Laurent. (1997). *Monk*. Translated by Jonathan Dickinson. New York: Marlowe & Compnay.

Dixon, Steve. (2007). *Digital Performance: A History of New Media in Theater, Dance, Performance Art, and Installation.* Cambridge: MIT Press.

Glinksy, Albert. (2000). *Theremin: Ether Music and Espionage*. Chicago: University of Illinois Press.

Horn, Delton T. (1992). *Electronic Components: A Complete Reference for Project Builders*. Harrisburg: TAB Books.

Igoe, Tom. (2007). *Making Things Talk*. Sebastol: O'Reilly.

Igoe, Tom and Dan O'Sullivan. (2004). *Physical Computing: Sensing and Controlling the Physical World with Computers*. Boston: Thomson Course Technology.

Kahn, Ashley. (2007). *Kind of Blue: The Making of the Miles Davis Masterpiece*. Cambridge: Da Capo.

Kettlewell, Ben. (2002). *Electronic Music Pioneers*. Vallejo: ProMusic Press.

Levitin, Daniel J. (2007). *This is Your Brain on Music: The Science of a Human Obsession*. New York: Penguin Group.

Petzold, Charles. (2000). *Code.* Redmond: Microsoft Press.

Pinch, Trevor and Frank Trocco. (2002). *Analog Days*. Cambridge: Harvard University Press, 2002

Audio *Recordings*

Anderson, Laurie. (1982). *Big Science*. Laurie Anderson. Original sound recording made by Warner Brothers. 3674

Carlos, Wendy. (1968). *Switched on Bach*. Wendy Carlos. Original sound recording made by Columbia Records. 7194

Eno, Brian. (1975). *Another Green World*. Original sound recording made by Island Records. IPS-9351

Articles

Ross, Alex. (2008). "Song of the Earth" *The New Yorker*, http://www.newyorker.com/reporting/ 2008/05/12/080512fa_fact_ross

Websites

Arduino. *Arduino - Tutorials.* http://arduino.cc/en/Tutorial/HomePage

Cycling '74. *Tutorials. cycling74.com/section/tutorials*

Create Digital Music. *Create Digital Music.* createdigitalmusic.com

Metropolitan Museum of Art, New York. *Musical Instruments*. metmuseum.org/works_of_art/ collection_database/musical_instruments/listview.aspx?page=1&sort=0&sortdir=asc&keyword=&fp=1&dd1=18&dd2=0&vw=1