A Novel Input Device For Thumb Control

Philippe S. Zaborowski

Music Technology Area
Schulich School of Music
McGill University
Montreal, Canada

August 31, 2007

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Arts.

2007/08/30

Abstract

This thesis project consists of developing a hand-held, one-thumb input device small enough to be placed on gestural controllers and providing performers with a simple way to select between multiple options with few movements of a thumb.

Most commercial input devices for the thumb make the assumption that buttons are the de facto standard. However, buttons are not always the best solution especially where space is very limited. This project explores a new approach to input design by analyzing the thumb's movements and designing a suitable input device that can track them. The design progression from the first to the last prototype is presented in great detail.

The final prototype is compared to other existing one-thumb input devices used in gestural controllers and mobile computing. The shortcomings of each one-thumb input method is described with the final prototype presented as a possible solution in each case.

Resumé

Cette thèse présente le développement d'une interface de contrôle spécifique pour le pouce. Elle et destinée à être montée sur un controleur gestuel pour rendre possible une selection simple de plusiers options ave peu de mouvements du pouce.

Les interfaces de contrôle pour le pouce que l'on trouve "dans l'industrie" se restreignent à l'utilisation de boutons. Cependant, notament dans le cas où l'espace est limitée, ceux-ci constituent pas toujour la meilleure alternative. Ce project explore donc une nouvelle approche pour la conception de telles interface en se basant sur une analyse des mouvements courants du pouce. Dans cette thèse la progression du design des prototypes pouvant capter ces mouvements sera abordé de manière détaillé.

Le prototype final est comparé à d'autres interfaces de contrôle pour le pouce disponibles sur des contrôleurs gestuels musicaux et aussi sur d'autres dispositifs portatifs (téléphones, lecteurs MP3 etc). Les problèmes de chaque interface sont décrits et il est montré comment le prototype final peut remédier à chacun d'entre eux.

Acknowledgments

I would like to thank my advisor Marcelo Wanderley for his understanding and help in completing this thesis project. I would also like to thank my family for their love and support.

Contents

List of Figures

List of Tables

List of Acronyms

FSR	Force-Sensing Resistor
T9	Text on 9 keys (a predictive text entry method for cell phones)
LED	Light Emitting Diode
IC	Integrated Circuit

Chapter 1

Introduction

Using parts of the body not directly involved in playing notes to increase an instrument's control possibilities has always been a trend in both instrument design and performance. For example, the trumpet is an instrument that only requires a single hand to play notes. In the 1920s, a technique pioneered by jazz musicians using plunger mutes was developed. The musician's free hand used a rubber plunger placed over the flare to filter the trumpet sound creating a "wah-wah" effect. This effect was also used extensively in electric guitars except that it was done electronically with the use of a pedal. More recently, with the availability of a wide variety of small sensors, the trend has been to attach them directly to the instrument and use any free fingers or thumbs to actuate them in order to increase the instrument's control possibilities [3] [16].

In many handheld musical controllers the fingers are occupied with playing musical notes, while the thumbs are involved in supporting the device and also in controlling program changes, octave shifts, or changing other musical parameters. In these devices, the thumb usually actuates various switches placed in a row [25] or distributed in matrices [4] [23]. But the thumb's range of motion is limited and the available space on these controllers is too small to accommodate a large number of switches. Also, the lack of visual feedback is a limiting factor to augmenting the number of switches available.

A number of different schemes have been devised to compensate for this. One approach is to use potentiometers or rotary encoders to scroll through a list of dozens of different parameters [9] [20]. This technique is not very practical because it requires fine tuning by the thumb with practically no tactile feedback. Another approach actually involves the fingers as well, which bars them from being able to play notes during the process [25]. Therefore, there is a need for an input device with a very small footprint that gives a single thumb the ability to make dozens of unambiguous selections without the need for visual feedback.

1.1 Project Overview

This thesis presents the design of such a device. The design process began with observing the movements of the thumb. The thumb can do three independent movements: a horizontal movement (left/right), a vertical movement (up/down), and a flexing or extending movement of the thumb (flex/extend). The next step was designing an input device using suitable sensors that can track these movements. Five prototypes were built each one improving on previous prototype's weaknesses.

Many possible applications of this input device were examined. Musical applications included both traditional MIDI wind controllers as well as alternate gestural controllers. Perhaps one of the more intriguing possible applications was in text entry. The potential of using the input device for text entry in mobile computing was presented.

1.2 Thesis Overview

The remainder of this document is organized as follows. Chapter 2 (background) describes the problem and analyzes design solutions that were presented by others. Chapter 3 describes the design process for the input devices developed for this thesis and the rationale behind the decisions. Chapter 4 discusses possible applications of these input devices. Finally, Chapter 5 presents conclusions, and points to future work.

1.3 Contributions

The main contributions of this thesis are the exploration of issues related to one-thumb input devices, their design and construction as well as their application in music and text entry.

Chapter 2

Background

This Chapter is divided into two main sections. The first section examines musical controllers that incorporate one-thumb control. The second section examines commercial one-thumb input devices for text-entry.

2.1 Musical Background

In a majority of musical controllers and acoustic instruments augmented with sensors surveyed for this thesis, the thumb was used to control one-dimensional effect parameters like volume or distortion. In a couple of cases, the thumb was used for much more sophisticated purposes controlling a wide variety of discrete parameters. The former cases will be described first.

Burtner [2] describes a tenor saxophone augmented with sensors. In this case, FSRs are attached beside each of the thumb rests of the saxophone and the applied thumb pressure was used to control an effect parameter including distortion, frequency modulation, or the amplitude of noise generators. Kapur et al. [11] retrofitted a sitar with multiple sensors in order to extract gestural information such as pitch, thumb pressure, pluck time and head tilt. Unlike Burtner [2], Kapur et al. [11] first described how he analyzed the gestural information from a performing sitar player before deciding how to map the sensor data. From his analysis, Kapur et al. [11] deduced that the thumb pressure was best suited to control such parameters as resonance oscillation, volume, delay, and warping of partials. Sinyor and Wanderley [22] created a gyroscopic instrument with an FSR placed at the thumb position which was used to control volume or a delay. Similarly, Lebel and Malloch

[13] also used an FSR to control the volume with the thumb. Schiesser and Traube [21] had each thumb controlling both a switch and an FSR. The switches were used for triggering musical events while the FSRs were used to control harmonic distortion. Finally, Freed and Uitti [6] used an FSR strip accessible by the thumb to control the vibrato of a cello augmented by sensors. In all these examples the thumb's applied pressure was simply used to control a specific effect parameter.

Fig. 2.1 Yamaha's WX5: top view

Fig. 2.2 Yamaha's WX5: bottom view

Several commercially available wind controllers have devised schemes for the thumbs to control a wide variety of parameters. One of the more popular of these wind controllers is the WX5 (figure 2.1 and 2.2) by Yamaha [25]. The left thumb controls a group of four buttons used for changing octaves while the right thumb controls a pitch bend wheel and two other buttons, namely the key hold button and the program change button. The key hold button can be used to hold a specified note while playing other notes. Pressing down on the program change button changes the functionality of the note playing keys (controlled by the fingers) so that they can used for changing voices, bank number transmission, or the MIDI transmit channel. Using the note playing keys is a big disadvantage because the musician has to temporarily stop playing notes in order to make a voice change. Another popular model the EWI4000s made by Akai has a similar scheme for selecting a new voice [1].

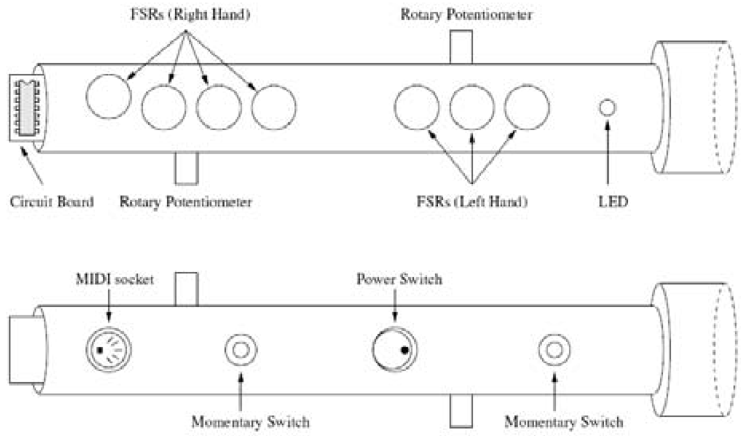

Fig. 2.3 The Pipe layout diagram: View from above (top) and below (bottom). Designed and built by Scavone [20]

The Pipe is an experimental MIDI wind controller designed and built by Scavone [20]. The top of The Pipe has an FSR sensor in each of the seven drilled finger depressions in a traditional two-hand wind instrument arrangement. The bottom has two momentary switches and each side has a rotary potentiometer (figure 2.3 and figure 2.4). In one configuration of The Pipe, the rotary potentiometers were used to circulate through MIDI program numbers 0-127. However, this was not deemed an adequate solution because it requires the thumb to fine tune the potentiometer without any active tactile or visual feedback. Even with a scroll wheel it would be hard to know the exact value selected without a display. Another disadvantage is that it is not possible to 'jump' directly to a particular value without going through all the values in between.

Fig. 2.4 The Pipe: View from above (top) and below (bottom). Designed and built by Scavone [20]

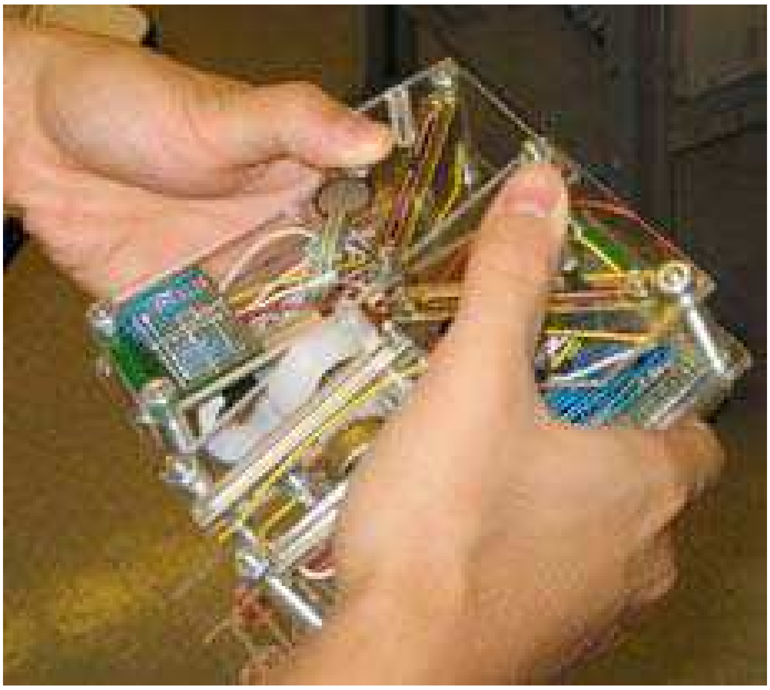

Fig. 2.5 The Bento-Box: designed and built by Hatanaka [9]. The fingers control the notes and effects while the thumbs control chord selection, volume, key signatures, and style selections.

The Bento-Box, a novel MIDI controller designed by Motohide Hatanaka, employs a similar approach to The Pipe [9]. The notes are played by pressing down on buttons using the fingers of the right hand while the effects are controlled by the fingers of the left hand. Instead of rotary potentiometers, the Bento-Box uses slide potentiometers to control various discrete parameter changes such as chord selection, key signature, and style selection. FSRs are also added for volume control along with a couple of buttons used in conjunction with the potentiometers (figure 2.5 and figure 2.6). However, this approach was abandoned presumably for the same reasons as the The Pipe's rotary potentiometers. Figure 2.7 shows another example of a one-thumb input device comprising buttons and a slide potentiometer while figure 2.8 shows a thumb controlling it.

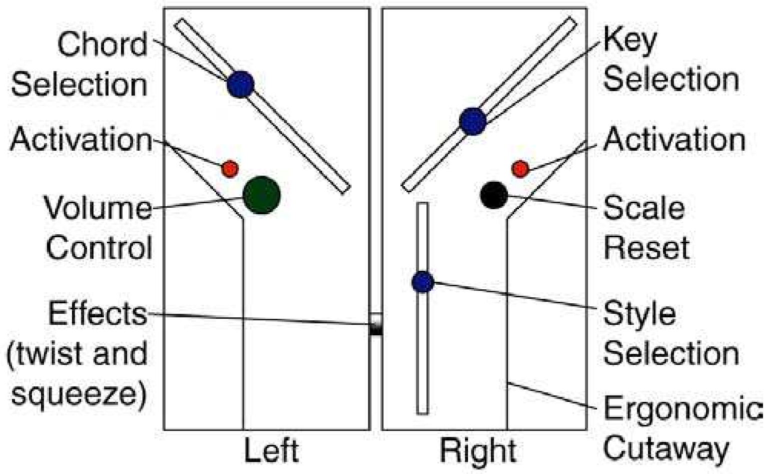

Fig. 2.6 The Bento-Box layout diagram for the thumb controls.

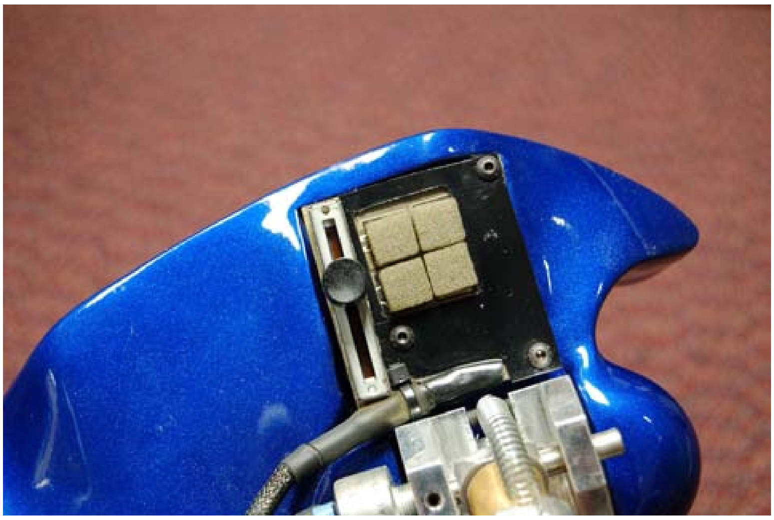

Fig. 2.7 The Meta-Instrument: 4 switches and a slide potentiometer used for thumb control.

Fig. 2.8 The Meta-Instrument: the thumb is actuating the input device.

2.2 Text Entry Background

This section describes thumb input devices used in common commercial products like cell phones and handhelds. A brief summary of the text entry methods will be presented as well as some of their limitations.

2.2.1 Cell Phone Keypad

The most common mobile interface is the cell phone's 12-key keypad. Although fine for entering numeric characters, entering letters requires a more elaborate method to compensate for the fact that there are only 12 keys to choose from.

Fig. 2.9 The cell phone keypad.

The 12-key keypad layout is well known to anyone that has a regular telephone or a cell phone. There are 10 number keys with a star (*) key and pound (#) key. Three or four letters are added to the number keys 2 through 9. When the user is inputting text via Multitap on the cell phone, a particular character is typed by locating the key with the desired letter and pressing down on it one or more times until it cycles into view. For example, typing the letter "S" would require the user to press the number 7 key four times. The advantage of this is that it is simple and comprehensible, but the need for multiple key presses for a single character considerably slows typing. Another example demonstrating the limitations of this method is typing the word "ON". This would require pressing the 6 key three times for the letter O, and twice for the letter N. In other words, the user would press the same key 5 times in a row and it would be impossible for the cell phone's software to know if the user intended to select the letter O first followed by the letter N or vice versa. This problem, known as segmentation, is solved by pausing after the first character and waiting for the system to timeout. Another segmentation technique involves pressing a special key between the O and the N in order to separate them. In summary, the Multitap method can be quite tedious and completely impractical for doing text editing or even writing long emails.

Fig. 2.10 The miniature qwerty keyboard.

2.2.2 Miniature Qwerty Keyboard

The miniature qwerty keyboard is currently one of the most common interface for PDA and smartphones. PDAs and smartphones are beginning to use more sophisticated software like text editors and web browsing. These types of devices are not designed for single thumb input like cell phones and text entry is usually done with both thumbs while the palm and fingers grasp the device. More recently, Apple released the iPhone that has a soft keyboard. Instead of using buttons, the keys of the qwerty keyboard are displayed on a touch screen. The advantage is that the soft keyboard can disappear when text entry is not necessary, doubling the size of the display area. However, not having actual buttons means that there is no tactile feedback.

2.2.3 Related Research

Dunlop and Crossan [5] jointly developed a predictive text input scheme that works similarly to T9. They compared it to Multitap and found that their predictive technique used half the keystrokes and was about 20% faster.

In a very intersting paper Kober et al. [12] compared ambiguous and disambiguous predictive techniques. Their research showed that as users make typing mistakes, the speed of disambiguous predictive techniques degraded to Multitap levels or worse.

Gutowitz [8] came to similar conclusions in his 2003 paper. Predictive techniques were found to be faster than Multitap only if the words were in the dictionary and spelled correctly. If these conditions were not met, predictive techniques would perform significantly worse.

Pavlovych and Stuerzlinger [18] described the Less-Tap technique that is the same as Multitap except that the letters are rearranged within each button according to their frequency. Surprisingly, the Less-Tap technique only resulted in an average speedup of 9.5%, although some participants were able to type 30% faster than Multitap.

Inspired by T9 and the alphabetically constrained cell phone keypad, Gong and Tarasewich [7] presented a research paper that explored and compared different constrained keypads with unconstrained ones. Their conclusion was that constrained keypads with a predictive scheme were easier to learn for novices, but that the optimal number of keys and configuration of the letters depended on the words tested for in the study.

Wigdor and Balakrishnan. [24] compared their own text entry technique called Chordtap with Multitap. Chordtap is a technique that used one thumb to select the cell phone key with the desired letter, while the other hand presses one of three switches located on the back of the phone to select the particular letter on the key. The technique was found to be faster than one handed or two handed Multitap.

Two input devices for text entry, Multitap and Rollpad, were compared in Oniszczak and Mackenzie [17]. Similarily to Chordtap, Rollpad's goal was to be able to both simultaneously select the key with the desired letter, and at the same time select the particular letter on the key. Rollpad accomplished this by using the inclination of the keypad while the key was depressed.

Chapter 3

Prototypes

This chapter presents the 5 prototypes designed and built for this thesis , and shows how the process was improved, from the first to the last. The physical apparatus, sensors, electronics and software are described in detail and their advantages and disadvantages are discussed as well. The chapter begins with the design rationale behind the prototypes.

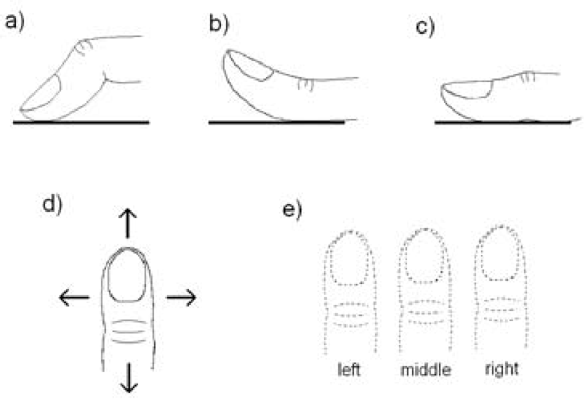

Fig. 3.1 *Touching states*: a)tip b)heel c) flat. d) *Pushing states*: North, South, East and West. e) *Lateral stroke states*

3.1 Design Rationale

The first step in the design process is the analysis of the movements of the thumb. It was observed that the thumb can touch a surface in three different ways namely with the tip (figure 3.1 a)), with the “heel” (figure 3.1 b)), or flat (figure 3.1 c)). These three touching movements were classified as *touching states*. It should be noted that the number of touching states can be increased depending on the input device. The thumb can also push North, South, East or West. These 4 movements (figure 3.1 d)) were classified as *pushing states*. Pushing states are when the thumb pushes on a stationary object like an isometric joystick. If we were to allow for pushing the joystick diagonally, there would be 4 extra pushing states. However, none of the input devices described below have these added pushing states. Figure 3.1 e) shows the final category of *lateral stroke states*. In this case the thumb slides laterally (side-to-side) along the surface and has the option of beginning in either the left, middle, or right positions and ending in either the left, middle, or right positions. Therefore, there are 9 (3 X 3) different lateral stroke states. The combination of these different states served as the inspiration for the various designs described below. For example, because the lateral stroke states are largely independent of the lateral stroke states, there are 27 possible combinations of lateral stroke states with the touching states (3 touching states X 9 lateral stroke states). One possible combination could be a tip touching state starting in the middle and sliding along the surface of the input device to the right position. As we shall see in the first prototype, the number of touching states can actually be increase to six with the right sensor configuration.

3.2 First Prototype

3.2.1 Mechanical Apparatus

This first prototype uses combinations of 6 touching states with the 4 pushing states potentially giving a total of 24 possible combinations or *choices*. This prototype was made up of three switches in a row and one miniature joystick on top of the middle switch (figure 3.2). Figure 3.3 shows a hand holding the input device. A thin transparent sheet of acrylic was used for the base of the input device. The three switches, which are not visible on the figure, were fastened to the sheet of acrylic. Each switch had a transparent acrylic blocks fastened on top. An extra white block (panel) was placed on the front and back switches to elevate the structure relative to the joystick. These three blocks were held together by eight supporting blocks surrounding them (three on each side and one at the front and back). The contacting vertical sides of the blocks were lined with Teflon tape so that they moved independently of one another when the thumb pressed down on them. The joystick's wires can be seen coming out on the right side figure 3.3).

Fig. 3.2 First prototype.

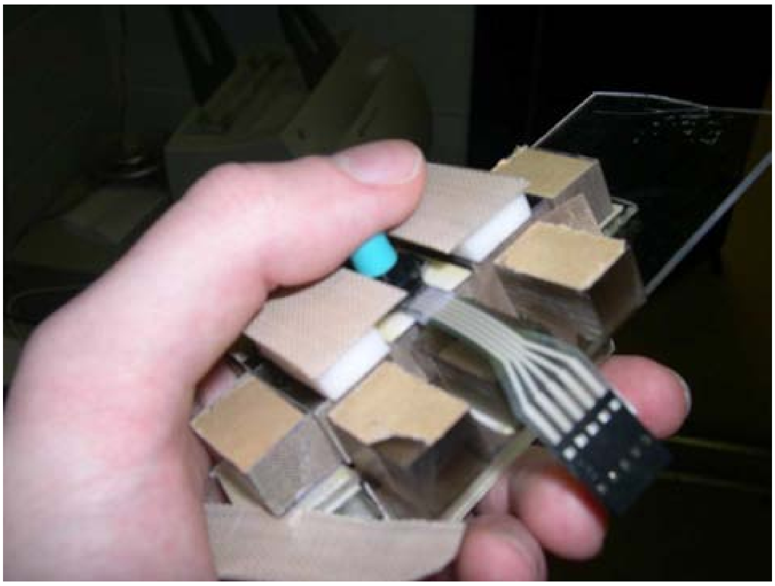

Fig. 3.3 First prototype being held.

3.2.2 Sensors and Electronics

The miniature joystick used in this prototype [10] was made by Interlink Electronics. Pressure exerted by a finger was transmitted to four FSRs located to the south, north, east, and west of the joystick's base. Four of the five wires, coming out from the joystick, outputted the analog FSR voltage signals using a simple voltage divider [10] while the fifth supplied the power to the joystick. Finally, the switches used in this prototype were tactile momentary (N/O) switches made by Omron Electronics with an operating force of 130 g. The analog joystick signals and switch signals were input into the computer via the Digitizer.

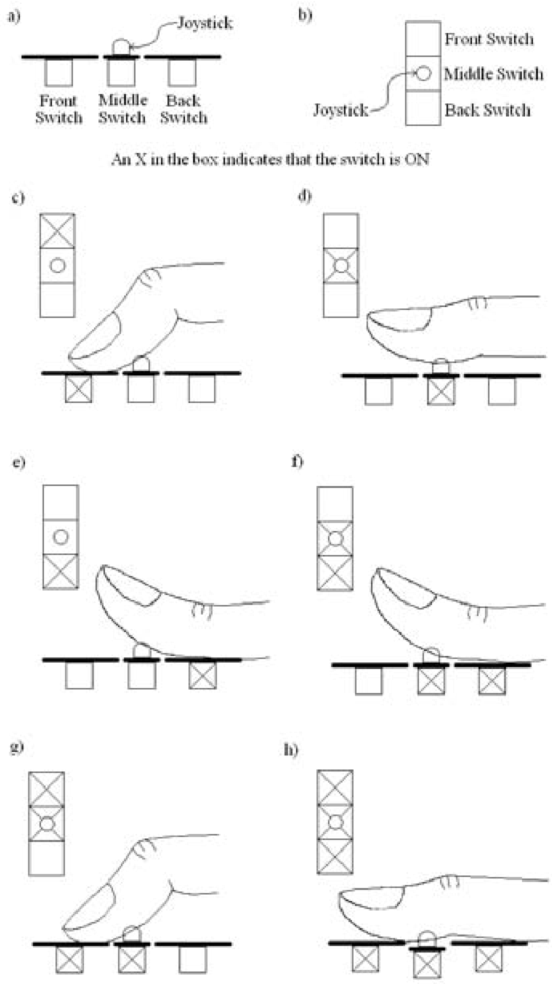

Fig. 3.4 First prototype: touching states.

3.2.3 Operation

Figure 3.4 a) shows a side view drawing of how the switches are activated in the first prototype. As mentioned above, one switch was placed in front of the joystick, the other behind, and a final switch was placed beneath the joystick. The thick black horizontal lines were the panels that were placed on top of the push button switches. The switches are represented by the boxes below the panels. An "X" inside the box indicates that the switch is actuated by the thumb (figure 3.4 c) to h)). Figure 3.4 b) shows a top view of the first prototype. The three squares are the panels. Figure 3.4 c) through h) shows all six

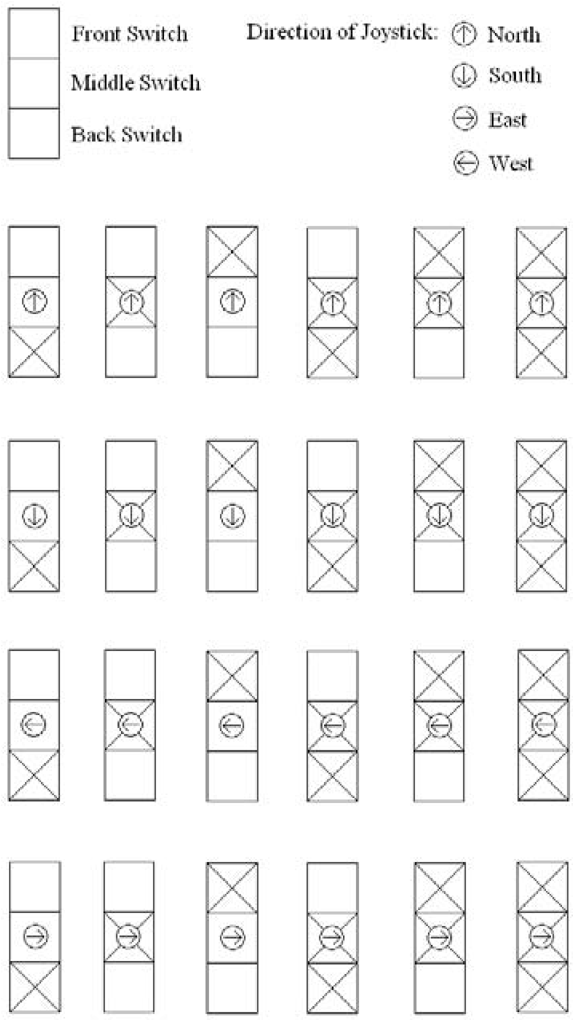

Fig. 3.5 First prototype: 24 different choices.

possible switch combinations that could have been simultaneously actuated by the thumb. Both a top view and a side view are shown for each of these combinations. It is important to note that the thumb is always in contact with the joystick. While actuating the front switch (figure 3.4 c)), the finger can simultaneously push the joystick North, South, East or West. As mentioned above, this is because the pushing states and the touching states were assumed to be largely independent of one another. When the joystick direction was factored in, the number of joystick/switch choices is 24 (figure 3.5).

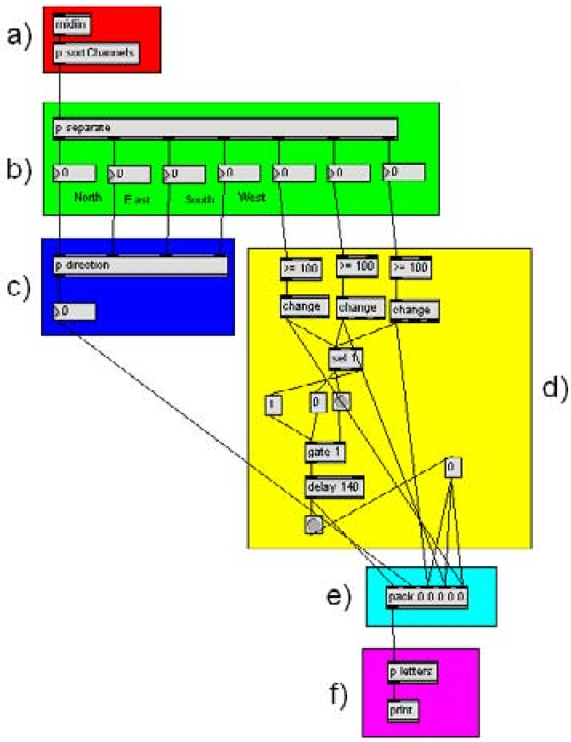

Fig. 3.6 First prototype: Max/Msp patch.

3.2.4 Software

As mentioned above, Max/MSP was used to analyze the incoming joystick and switch sensor data and output the recognized user's input. Figure 3.6 shows the patch used to do this. A subpatch called sortChannels was written (figure 3.6 a)) to input the raw data from the digitizer. The joystick sensor data (figure 3.6b)) was analyzed in the "direction" subpatch (figure 3.6c)) by simply outputting which direction signal (either North, South, East or West) was strongest at the time. This value was stored in the vector in figure 3.6 e) along with the actuated switches. Once the user lifts his/her hand off of the switches, the vector is sent to figure 3.6 f) where the choice is recognized. In other words, the joystick direction stored in the vector was the last direction before the thumb disengaged from the switches. This could be a little tricky because one would have to make sure one was applying pressure on the joystick in the correct direction before lifting the thumb off of the switches.

3.2.5 Discussion

This prototype met the basic requirements outlined in the beginning of this document, namely the user was able to perform pushing states and touching states at the same time. But the use of a joystick proved problematic because one would inadvertently apply unwanted pressure on the isometric joystick during the act of actuating the switches which would go completely unnoticed by the user. In other words, the assumption that the pushing states and the touching states are largely independent was proven incorrect. The raised joystick also made selecting the front or back switches somewhat awkward. Moreover, because the joystick was raised higher than the front or back switches it was easy to sometimes unintentionally trigger the middle switch underneath it. Finally, the joystick would sometimes be inadvertently pushed in the wrong direction during the touching state movement. For all its shortcomings, the first prototype did work if the user was careful enough. Furthermore, it was assumed that most of these problems were solvable and a new design was conceived. In the end, this prototype hinted at the possibility that a one thumb input device based on the idea of touching states and pushing states could be designed if enough care was taken to tackle these problems.

3.3 Second Prototype

3.3.1 Mechanical Apparatus

In order to improve on the first prototype, the joystick was attached on top of a spring mounted slider (male part) that would retract downward into the slider guide (female part) and out of the way when the user would actuate the front or back switches. Figure 3.7 shows the red joystick on top of the slider with switches on each side. Although it was possible to add a middle switch that would be triggered at the end of the spring mounted slider's travel, none was added. If the second prototype showed enough promise it would be added later.

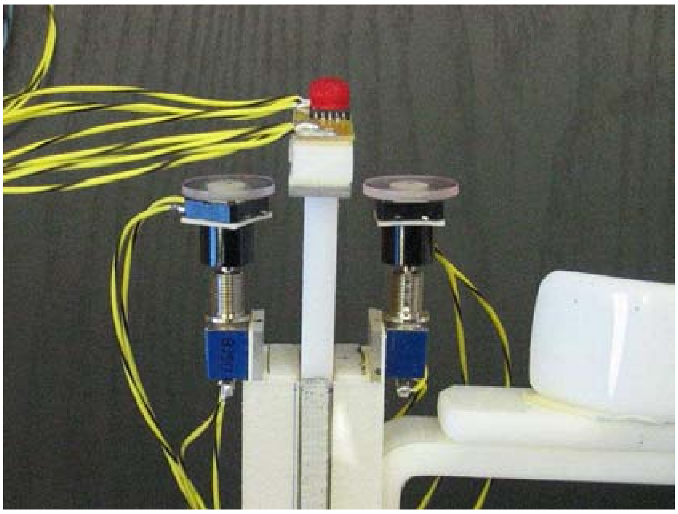

Fig. 3.7 Second prototype.

Another difference was that there were two switches staked on top of the other. The same tactile switches were used as in the first joystick prototype but they were actually mounted on top of regular pushbutton switches. According to the specifications these pushbutton switches would require three times as much force to activate them and thus there would be two different pressure levels that could be detected by the stacked switches, increasing the number of possible choices. However, this idea was immediately abandoned because it was awkward to have the user distinguish between two pressure levels. A hot glue gun was used to immobilize the pushbutton switch that ended simply being used as a platform for the tactile switches. Thus, there were only 3 touching states to choose from and the same 4 pushing states as in the first prototype. A round plastic panel was added to the top of each tactile switch to increase its contact area. A plastic handle was added to allow the prototype to be firmly held (figure 3.8. Teflon tape was adhered to both the slider and the inside of the slider guide.

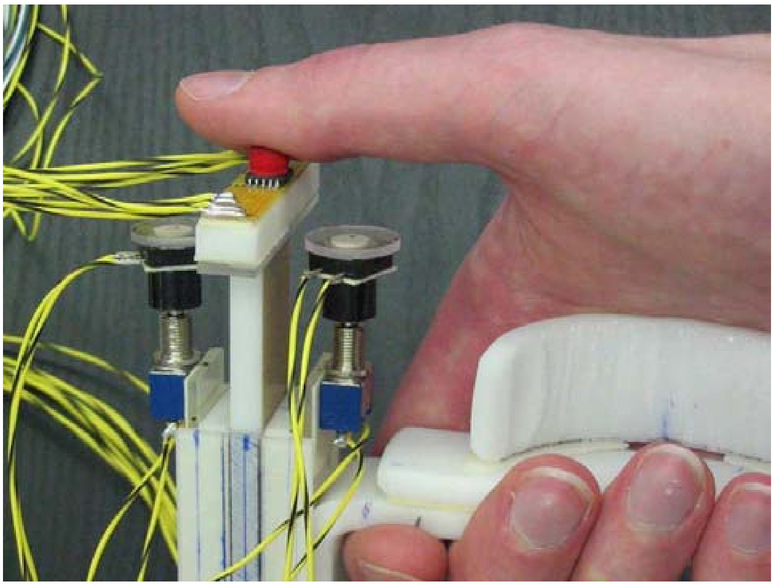

Fig. 3.8 Second prototype being held.

3.3.2 Sensors and Electronics

The CTS 109 Series joystick manufactured by CTS Corporation replaced the one from the first prototype. Pressure exerted by a finger on the joystick is transmitted to four strain sensitive thick-film resistors inside. A Wheatstone bridge and amplifier allowed for the output voltages to be proportional to the applied force which was an advantage over the non-linear FSR based joystick of the first prototype. The CTS joystick was not only able to detect lateral pressure along the x and y axis, but was also able to detect downward pressure (z axis) as well. However, only the x and y directions were used. A red rubber cap was placed on top of the joystick.

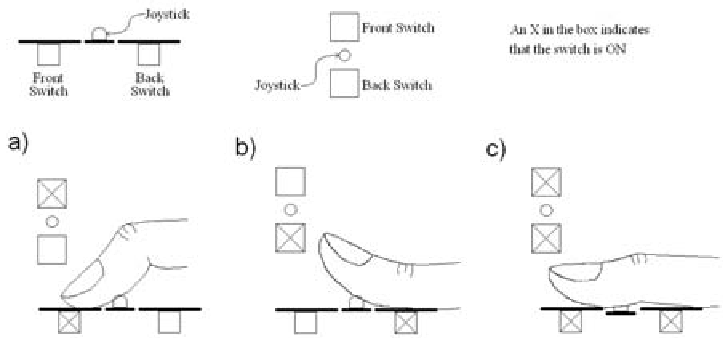

Fig. 3.9 Second prototype: touching states.

3.3.3 Operation

Figure 3.9 shows the 3 touching states. As mentioned above there is also 4 pushing states giving a total of 12 possible choices (figure 3.10). The joystick's plastic cap had a rough rubber texture keeping it firmly in place under the thumb without any slipping. The spring mounted slider retracts quite smoothly and nicely but fits snuggly inside the Teflon padded slider guide. The handle keeps the hand and by extension, the thumb, in perfect position above the joystick. There was nothing mechanically wrong with this second prototype and yet, as we shall see in the discussion, it did not meet the author's expectations either.

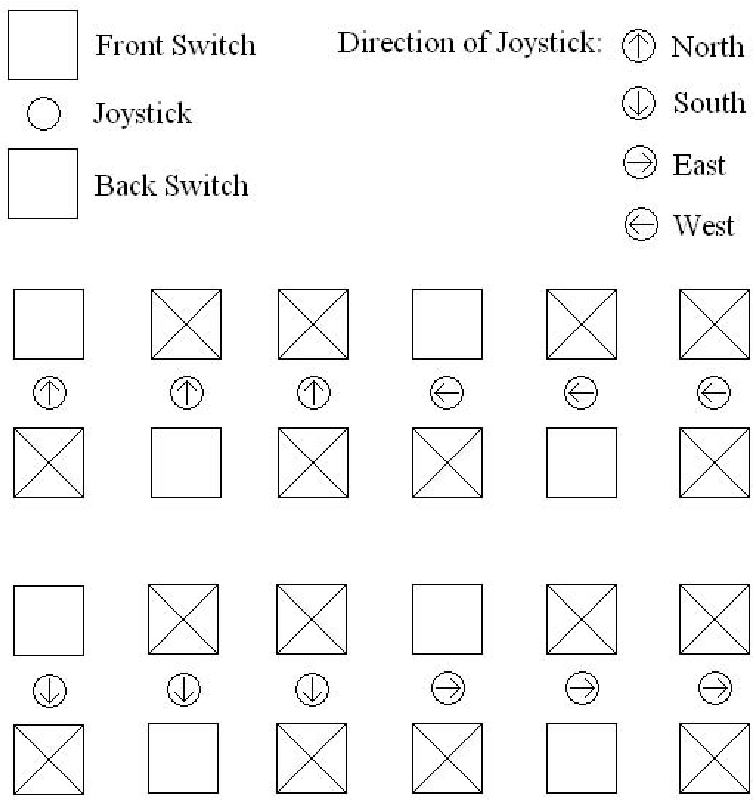

Fig. 3.10 Second prototype: 12 choices.

3.3.4 Software

The Max/MSP patch (figure 3.11) was very similar to the first except for a few details. The “p separate” (figure 3.11 b)) object had six outputs with the first and second being the x and y components of the joystick and the next two relating to the tactile switches. the last two outputs represented the pushbutton switches which were not actually used. The joystick data from the first two outputs were then sent on to figure 3.11 c) where the signals were analyzed and a direction established and then stored in the vector in figure 3.11 e). The switches which were activated were also stored in that same vector. As in the first prototype, once the switches were released by the user, the data stored in the vector was analyzed by “p vector” and the choice made by the user was determined.

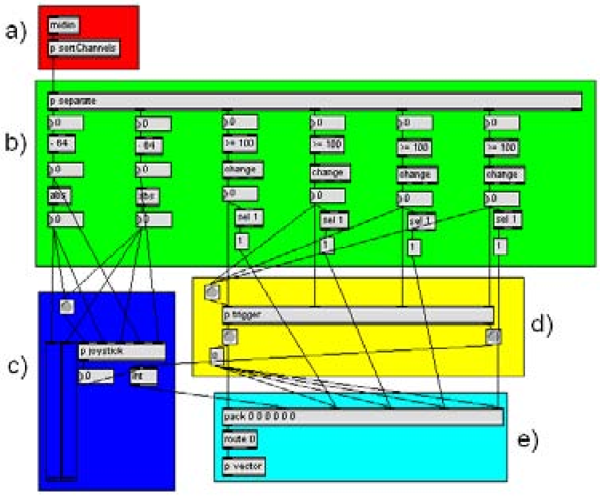

Fig. 3.11 Second prototype: Max patch.

3.3.5 Discussion

Although the spring mounted joystick made actuating the switches more ergonomic, surprisingly it did not solve the problem of inadvertently pushing the joystick in a wrong direction. This would occur because the user would simultaneously have to push the joystick and actuate a switch at the same time. For example, if the user wanted to push the joystick West and actuate the bottom switch, the switch actuation movement of the thumb would inadvertently push the joystick North as well, resulting in a North-West direction of the joystick which is an ambiguous direction. The isometric joystick's lack of tactile feedback made this prototype frustrating since the user could never really be sure if he/she was doing the correct movement. Thus, the joystick concept had to be scrapped and a new design had to be attempted.

3.4 Third Prototype

3.4.1 Mechanical Apparatus

After the unsatisfying performance of the joystick prototypes a new design incorporating touch sensitive strips like the one in figure 3.12 was devised. This prototype used the concept of touching states and lateral stroke states and had no pushing states. Two sliders, a front slider and a back slider, were both glued to a base (figure 3.13). Each base (with the strip) was fastened to a slider guide (figure 3.14) which in turn was placed onto the slider. The slider itself was fused to the main base of the prototype. Figure 3.15 shows a touch sensitive strip fixed to a slider guide and mounted on to the back slider with the front slider exposed. Unlike the second prototype, in this design the sliders were stationary while the slider guides moved up or down. Figure 3.16 shows a piece of plastic protruding from the side of one of the slider guides. The whole slider guide/strip assembly was actually resting on top of the switch, and thus would only move down as much as the travel of the switch which was half a millimeter. Once again, Teflon tape was placed on both the sliders and on the inside of the slider guides to minimize friction.

Fig. 3.12 Third prototype: pressure sensitive strip from Infusion Systems.

3.4.2 Sensors and Electronics

The strips mentioned above were bought from Infusion Systems (figure 3.12) and two tactile switches by ITT Industries were purchased from Digikey. The active area of the strip measured 14.0 cm by 2.0 cm and 0.6 cm thick and plugged directly into the digitizer.

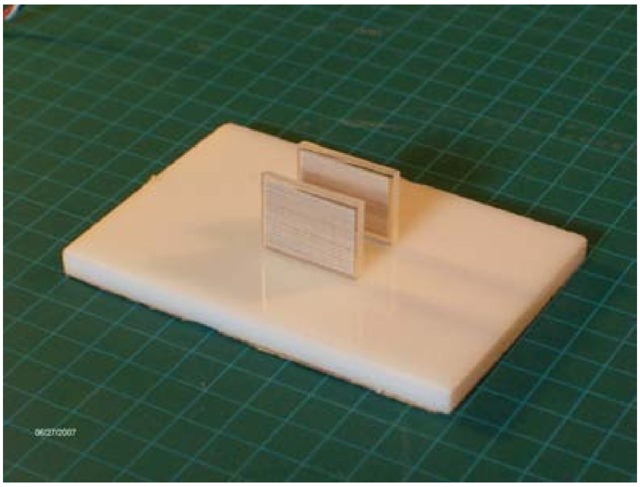

Fig. 3.13 Third prototype: male part.

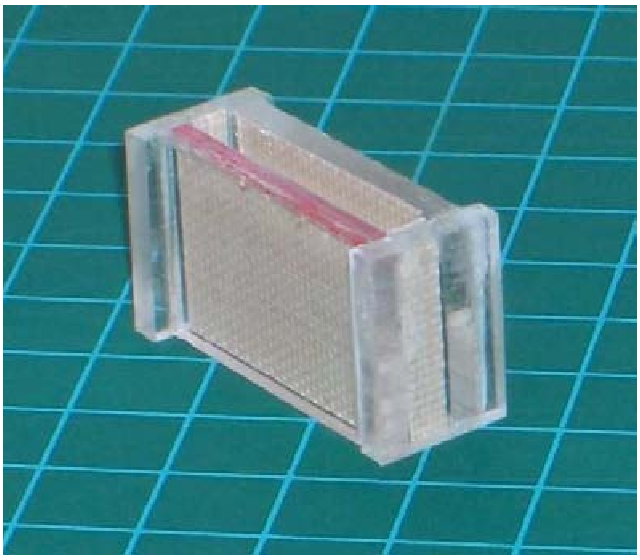

Fig. 3.14 Third prototype: female part.

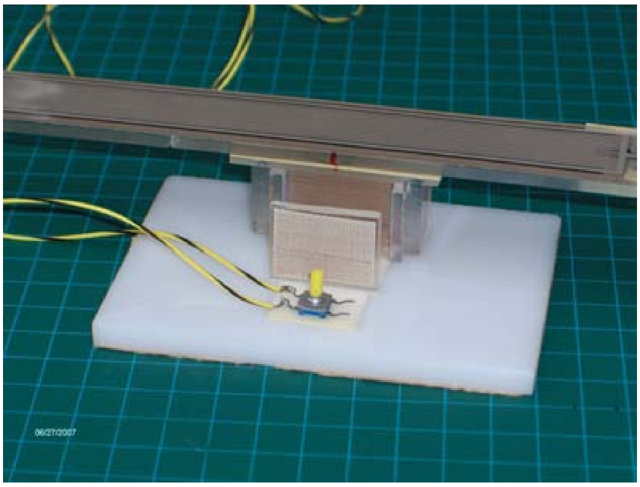

Fig. 3.15 Third prototype: front slider exposed.

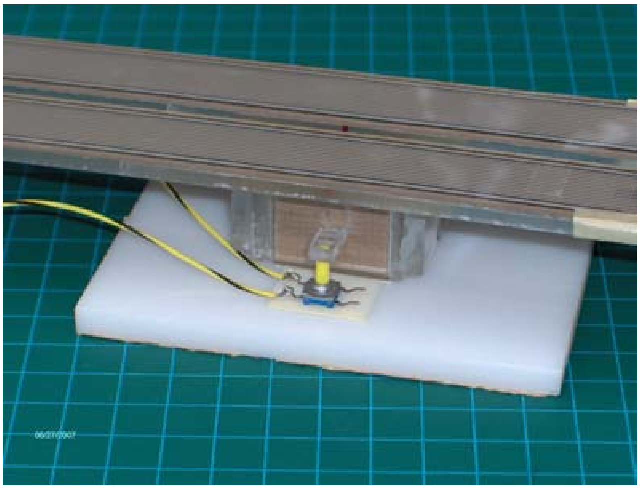

Fig. 3.16 Third prototype: slider guide and strip assembly actuating switch

The strip was a one dimensional sensor designed to measure the position of a finger along the surface of the strip. Although these sensors were supposed to be linear in fact they were found to be of very poor quality with each strip having a different maximum output voltage varying as much as 35% from one strip to the next.

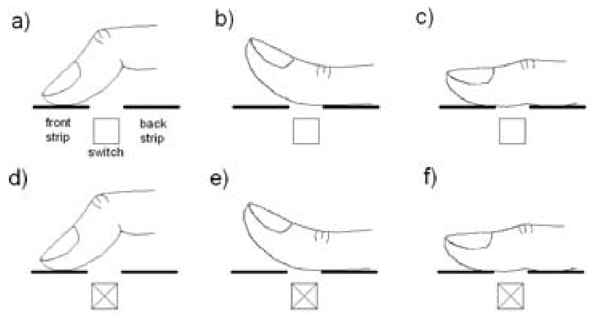

Fig. 3.17 Third prototype: 6 touching states.

3.4.3 Operation

The thumb can touch or slide along one or both sliders and can also actuate the switches by pressing down on one or both strips (figure 3.19). Therefore there is 6 touching states (figure 3.17) and 9 lateral stroke states giving a total of 54 choices (figure 3.18). Each of the 54 rectangles in figure 3.18 represent a top view of the third prototype with the top and bottom halves representing the front and back strips respectively. The arrows represent finger strokes along the strip where the tail of the arrow is the point where it first engages the strip and the head of the arrow is the last point on the strip before the finger is lifted off. The square dots symbolize the finger touching the strips without sliding. Finally, the red (or gray) and black colors represent whether or not the switch is actuated.

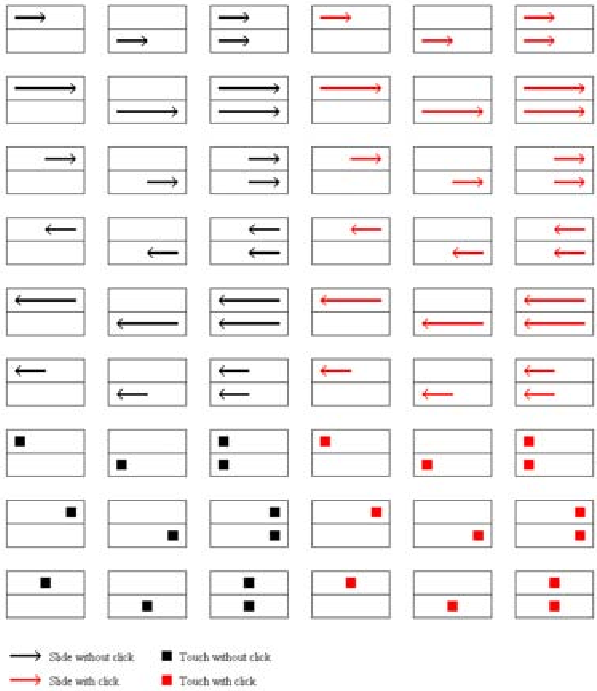

Fig. 3.18 Third prototype: 54 choices.

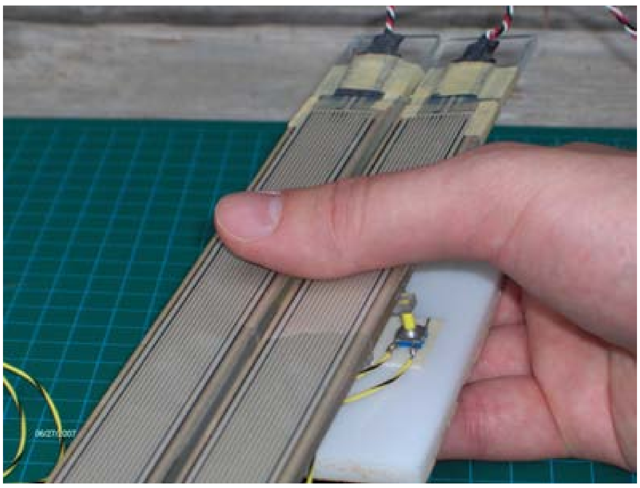

Fig. 3.19 Third prototype: thumb operating prototype.

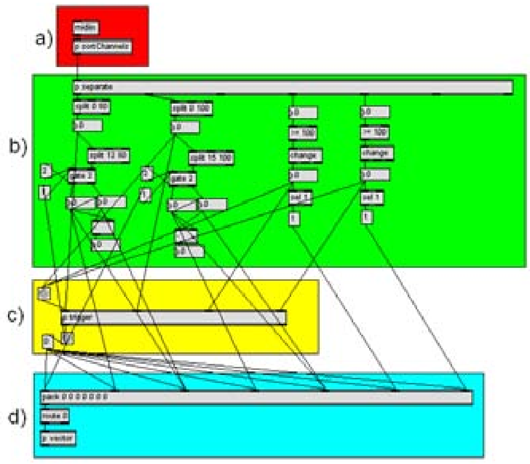

Fig. 3.20 Third prototype: Max patch.

3.4.4 Software

Figure 3.20 shows the Max/MSP patch used for this prototype. The first two outputs of "p separate" (figure 3.20 b)) are the touch sensitive strip sensor data, while the next two outputs represents the two switches. The patch is designed to capture the initial and final positions along with the actuation status of the switches. All of this is stored in the "pack" object in figure 3.20 d). When the finger is lifted off the strips the stored information was triggered (figure 3.20 c)) and sent to "p vector" where it is analyzed and the user selected choice is determined.

3.4.5 Discussion

Although this prototype fell a bit short of expectations, it showed the most promise of all the three prototypes. With the right sensors this design was the most likely to succeed. Nevertheless there were some problems. The biggest one was the touch sensitive strips required the thumb to apply a relatively strong and steady pressure during the entire stroke. This would often result in inadvertent triggering of the switches and using switches with a higher actuation would make pressing down on them too much of a chore. Another issue was the inconvenience of sliding along two strips at the same time. So even though theoretically it was possible to have 54 different choices, practically speaking, most were not very easy to do consistently or easily. Thus, if we were to eliminate all the problematic choices we would only be left with the first two columns of figure 3.20 or 18 choices. Friction of the finger along the surface was also problematic. Using Teflon tape helped a little but not enough. Using a switch for each strip was a silly design mistake since a single switch under both strips would have produced exactly the same number of choices. In other words, it would be practically impossible to slide along both strips while actuating one switch but not the other. The next prototype solved this problem by constructing the sliders as a single unit. In the end the touch sensitive strips were poorly made and not suitable for this application. A new sensor needed to be found as a replacement.

3.5 Fourth Prototype

3.5.1 Mechanical Apparatus

Once again, this prototype was based on the touching states and the lateral stroke states design concept. It was comprised of a base, two sliders projecting from the base (figure 3.21), and a slider guide structure, with two touch sensitive strips adhered to it, mounted on the two sliders. Figure 3.22 shows the bottom of the slider guide structure. Notice that there were three slider guides. The middle slider guide was used for the slider wheel assembly (figure 3.23 while the other two slider guides were mounted on top of the two the sliders projecting from the base. The slider wheel was made of a blue bead on an axle (so it can rotate)) installed on a slider. Figure 3.24 shows the slider guide structure mounted on top of the sliders, but without the slider wheel. Figure 3.25 displays a fully assembled fourth prototype held by a hand and shows the two touch sensitive strips as well as the spring mounted slider wheel. The slider wheel is simply used as a reference point for the thumb.

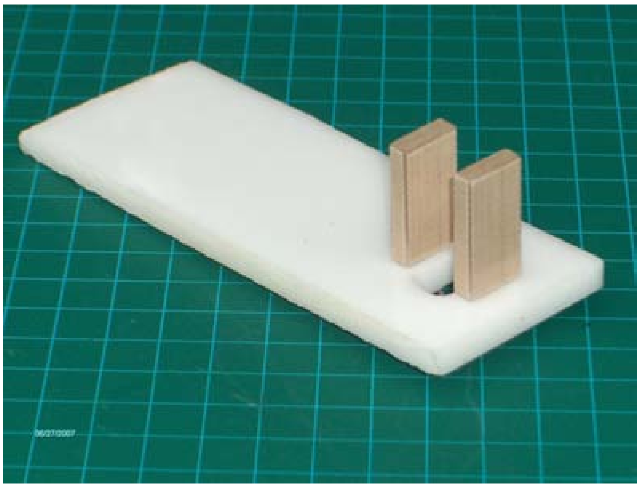

Fig. 3.21 Fourth prototype: male parts.

3.5.2 Sensors and Electronics

In this prototype, the touch sensitive strips were “homemade” by the author using an array of six FSR sensors for each strip (figure 3.24). The pressure sensors were placed in a row beneath the piece of soft polyurethane overlay with Teflon tape placed on top. Applying

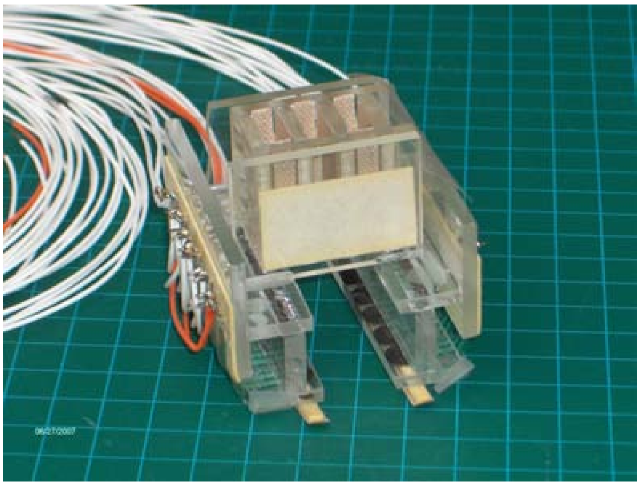

Fig. 3.22 Fourth prototype: view of the bottom.

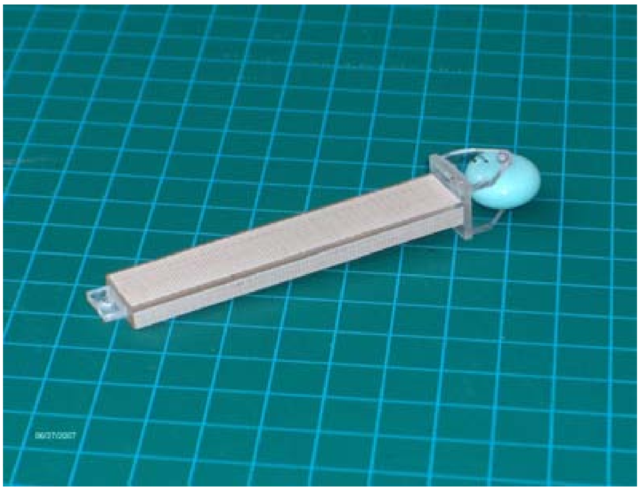

Fig. 3.23 Fourth prototype: slider wheel.

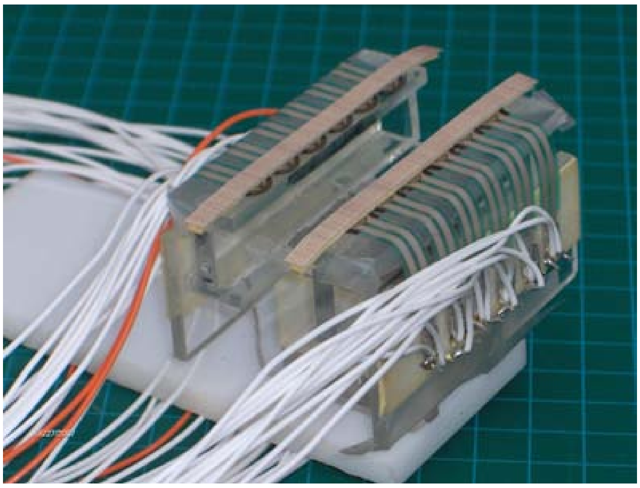

Fig. 3.24 Fourth prototype: without slider wheel.

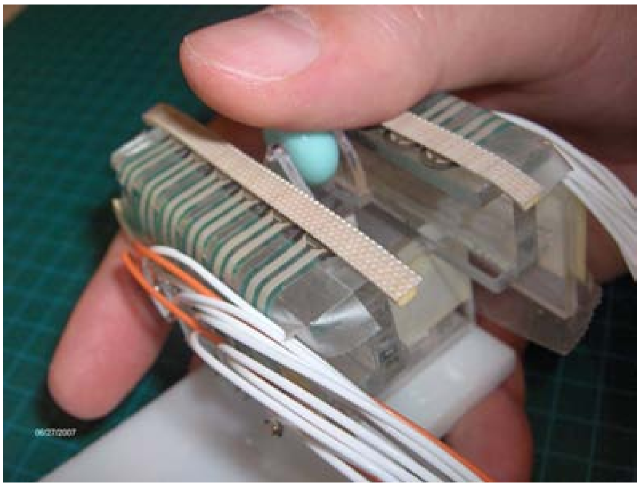

Fig. 3.25 Fourth prototype: thumb operating prototype

pressure on the overlay would deform it and transfer the pressure onto the pressure sensors below. As before, each sensor was connected to the Digitizer. Although it was possible to add a switch to this prototype, none was added for the preliminary evaluation. Because of previous experience, the author decided to quickly evaluate these new strips before proceeding any further with this prototype.

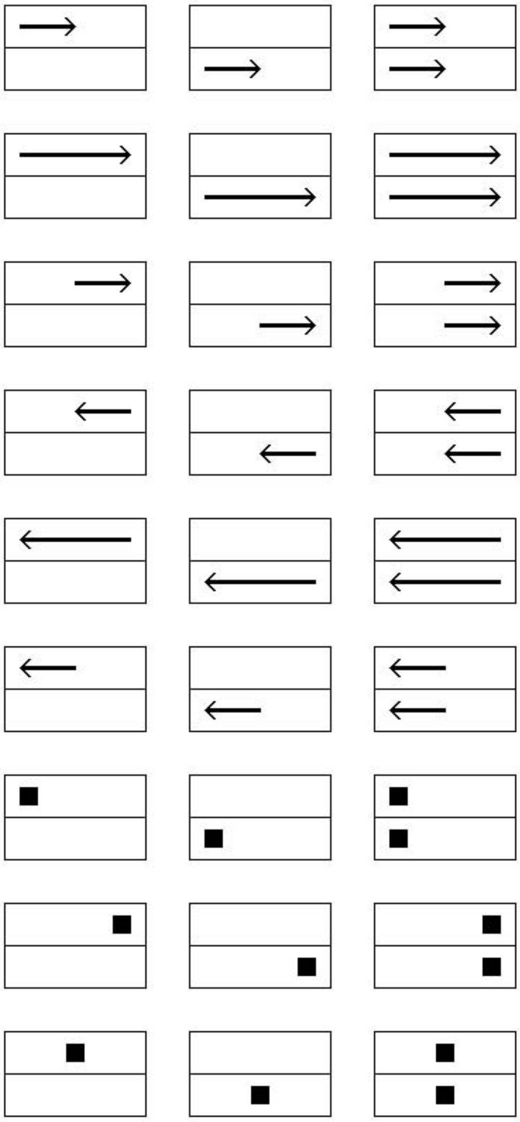

Fig. 3.26 Fourth Prototype: 27 choices.

3.5.3 Operation

The biggest structural difference between this prototype and the third was the slider wheel. It was used so that the thumb would be able to distinguish between the middle, left side and right side of the strip. Because there was no switch beneath the strips there are only 3 touching states and 9 lateral stroke states. Figure 3.26 represents all the possible choices available for the fourth prototype. As with the third prototype, the arrows represent a thumb sliding along a strip, while the squares represent touching the strip.

3.5.4 Software

The software used in this prototype was similar to the third prototype except the position along the "homemade" strips determined which FSR sensor in the strip had the highest value. Linear interpolation would have given a much more precise location, but this was not necessary because the important criteria was whether the position of the thumb was on the left, middle or right.

3.5.5 Discussion

Although, individually the FSR sensors were more sensitive than the infusion systems touch sensitive strip used in the previous prototype, when arranged in a row with an overlay they required even more pressure than the infusion systems strips. Moreover, sliding the thumb along the strips was fatiguing and not a realistic solution for effortless input. Sliding the thumb along both strips compounded the problem. The idea of using pressure sensitive sensors like the ones used in both prototypes three and four were abandoned. One positive result from this prototype was that the size and spacing of the strips felt right. With the right sensors, this setup had the potential to work.

3.6 Fifth and Final Prototype

3.6.1 Mechanical Apparatus

The final prototype uses capacitive technology and is simply made up of four green PVC insulated wires and a single red PVC insulated wire perpendicular to the green ones (figure 3.27). The wires are fixed to an acrylic sheet using double sided tape. The acrylic sheet itself is mounted on top of a tactile switch. The wires are connected to circuits on breadboards which, in turn, plug into the Digitizer (figure 3.28. For testing purposes some choices were mapped to letters of the alphabet.

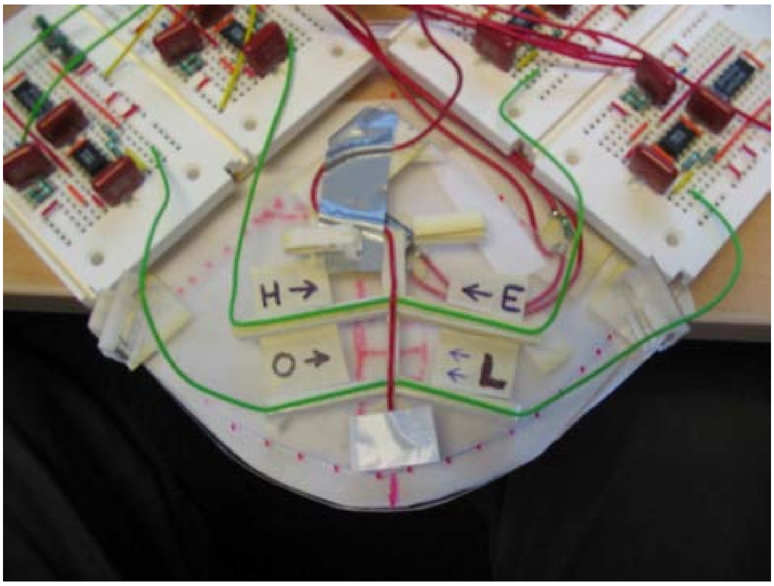

Fig. 3.27 Fifth Prototype: close-up view of strips.

3.6.2 Sensors and Electronics

The heart of the circuit (figure 3.29) is the QT301 made by Quantum Research Group which is a 8-pin DIP capacitance to analog converter IC [19]. Figure 3.29 represents a single electrode circuit (thus the prototype requires five of these) with the square electrode representing the wire electrode. The analog output is connected directly to the Digitizer. The tactile switch used in this prototype is the same as in the first prototype.

The QT301 essentially measures the amount of contact surface area between a thumb and the wire. The more contact surface area, the larger the analog output voltage. In

Fig. 3.28 Fifth Prototype: wider view of strips, breadboards, digitizer and computer.

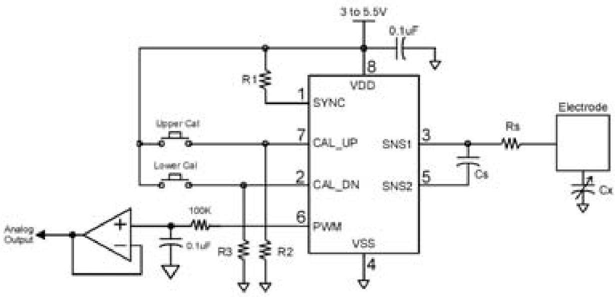

Fig. 3.29 Fifth Prototype: circuit.

order to have the circuit working properly, the QT301 needs to be calibrated to sense the amount of capacitance when the thumb is away from the wire, which is the lower bound, and when the thumb is pressed up against the wire, which is the upper bound.

Unlike FSRs, capacitive sensors are truly touch sensitive and are thus ideal for this application. Similarly to the third and fourth prototypes, the wire electrodes were arranged such that two green wires make up the front touch sensitive strip while the other two make up the back strip (figure 3.27). Even though the electrodes measure contact area, they were essentially used as on/off switches in this application. Thus, each strip could only sense three discreet areas namely when the finger was on the left green wire, or the right green wire, or on top of the middle red wire.

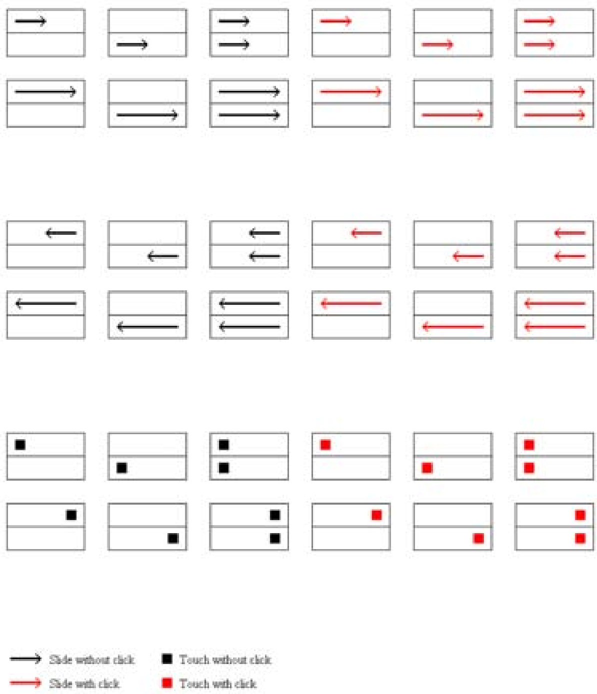

Fig. 3.30 Fifth Prototype: 36 choices.

3.6.3 Operation

Just like the third and fourth prototypes, this one is also based on the lateral stroke states and touching states. There are 6 touching states and although it is theoretically possible to make all 54 choices (see figure 3.18 the number was reduced to 36 (figure 3.30)) due to the fact that the middle red wire makes the margin of error very slim and it is difficult to consistently start in the middle position without inadvertently triggering the left or right wires. In other words, because the sensors were essentially on/off switches the user would have to make contact with the middle red wire before touching any of the other wires which was too difficult to do consistently. So all the choices in figure 3.18 that began in the middle position were eliminated and only the choices beginning on the left or right were used for this prototype.

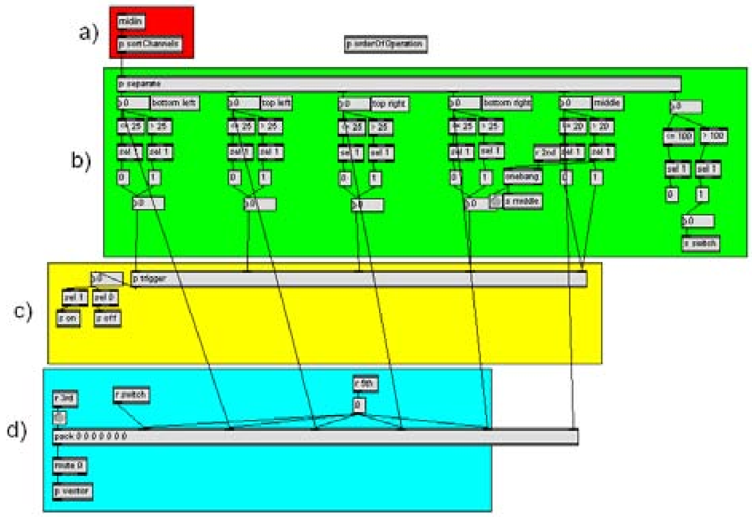

Fig. 3.31 Fifth Prototype: Max patch.

3.6.4 Software

The max patch used in this prototype is very similar to all the others (figure 3.31). A more complicated patch was designed to include all 54 choices but it was much more complicated and was not robust enough to handle very small mistakes. In other words, the input stroke would have to be perfect every time with no margin of error. As mentioned above, this was because the sensors were essentially on/off switches instead of being able to estimate the position of the thumb along the strip.

3.6.5 Discussion

This prototype was by far the best and worked quite well overall. Using truly touch sensitive strips made all the difference. Sliding the thumb over the plastic wire insulation was smooth and effortless and made inadvertent triggering of the switch much less likely. However, only using three sensors per strip made it practically impossible to include the lateral stroke states that begin in the middle position. But the design concept proved successful enough and an improved version using more sensors would be good enough to test in a usability study.

Chapter 4

Discussion

The goal of this research was to create a one-thumb input device giving the user the ability "touch type" dozens of different commands using a single thumb. This device would be applicable in a variety of situations that require the selection of dozens of different discrete choices. The progression from the first to the last prototype went through many changes in both the sensors used and the design concept. The final prototype was found to work well enough to warrant further research. The following sections will discuss the progression of the prototypes as well as some possible applications in text entry and music.

4.1 Prototype Progression

The idea of creating a one thumb input device for the thumb was much more difficult than anticipated. The original concept was inspired by video game consoles and the IBM Thinkpad laptop isometric joysticks (a.k.a pointing sticks). It was observed that the thumb would move too much when controlling the joystick and did not seem to have the ability to control anything else. The idea of using an isometric joystick instead seemed like an interesting idea. The thumb would still be able to push a joystick, but would not need to be displaced from its original position. Therefore, the thumb might be able to actuate switches while it is controlling the joystick. The first prototype used an FSR based joystick made by Interlink Electronics. It was found that this particular joystick required too much pressure for this application. But that was the least of the problems. The biggest one was the fact that the joystick was raised too high and made selecting the front or back switches awkward. The high joystick position also caused inadvertent actuation of the switch below.

There was also the problem of inadvertently pushing the joystick in the wrong direction when actuating the front or back switches. It turned out that in this particular setup, the touching states and pushing states were not independent of one another as it was assumed in the beginning. The switches, on the other hand, were very good and had the right actuation pressure. Nevertheless, the joystick concept was not discarded and certain improvements were devised to improve it.

The second prototype used the same type of isometric joystick used in older laptops as a pointing device. Initial tests showed that it was quite sensitive, consistent, and overall a good sensor choice. It was mounted on a spring mounted slider. A building technique was developed that made the slider fit perfectly into the slider guide. This was important, as the whole point was to make sure the joystick did not have any horizontal movement. The joystick needed to stay perfectly still, and only be allowed to move up and down. Using Teflon made this vertical movement very smooth and could be pressed down effortlessly by the thumb. A handle was added and positioned so that the thumb would be in the most ergonomic position possible. The addition of the slider and guide certainly made it less awkward compared to of the first prototype. However, the second prototype did not solve the fact that the touching states and the pushing states were not independent of one another. The whole concept of this input device was founded on their independence. Unfortunately, this was not solved by adding a slider and guide. Therefore, the joystick concept was completely abandoned and a new design was needed. A design where the touching states were truly independent was needed.

The idea of using two touch sensitive strips was introduced in the third prototype. Each strip was mounted on top of a slider guide. The strip and slider guide unit was then placed on a slider that itself was mounted on a base. After initial testing, the strips were found to be quite inadequate. They required the thumb to apply strong and steady pressure along the whole lateral stroke. If this type of consistent and steady pressure was not applied, the sensor would lose the position of the thumb as if it was lifted off the strip. Unfortunately, applying consistent pressure along two strips was practically impossible. However, this prototype showed promise and it was decided to continue to build more prototypes using the two strip design concept.

For the fourth prototype, it was decided to build a “homemade” pressure sensitive strip using an array of six FSR touch sensors. The FSR array was covered by a deformable foam overlay and Teflon tape was stuck to the foam to minimize the thumb’s friction along the strips but was surprisingly not as effective as anticipated. In the end, the “homemade” strips suffered from the same problem as before: they required too much force to operate. One feature that distinguished this prototype from the rest was the addition of a roller wheel that was placed between the two “homemade” strips, protruding above the surface of the strips. It was designed to retract downward when the thumb rolled over it and served as a reference point to distinguish between the three possible starting points of a lateral stroke state: namely: left, middle, and right. This worked quite well and was a good solution for giving the user tactile feedback on the thumb’s position relative to the middle. However, the roller wheel was not used in the final prototype because it made the prototype much too large. It is possible that roller wheel concept might return in future work if a more compact design is created. In retrospect, this prototype design is similar to that of a mouse, in that the mouse’s scroll wheel is also between two strips. Except the strips are just plastic buttons and the scroll wheel does not retract very much.

The fifth prototype was the one that worked best primarily because of the capacitive sensors chosen for the “homemade” strips, that in the third and fourth prototypes were built using pressure sensitive technology. Capacitive sensors require almost no pressure because they sense the surface area that the finger makes with the sensor. So the lightest touch would easily be sensed making them ideal for this application. The only problem was not the sensors themselves, but the way the sensors were arranged. As can be seen on figure 3.27 each strip was comprised of two green wires. Thus, the left position would be sensed by a strip if the left green wire was touched and the same would be true for the right position. However, the middle position would be sensed when the finger touches the single red wire perpendicular and between the left and right wires of either strip. This design was flawed because it did not give the user room to consistently hit the middle. If the thumb was slightly off target, it would inadvertently touch one of the green wires. This could easily have been avoided if the wires were arranged in an array with each wire seamlessly merging with the other, making a single wire. In other words, the strip would be made up of three wires line up in a row.

Of course, using more sensors would be better. Five capacitive sensors per strip would probably be more than enough and give the user enough “wiggle” room to select the left, middle, or right positions he/she intended. Besides the low number of sensors, this prototype worked well enough and the problem of inadvertently actuating the tactile switch between the strips was substantially reduced. One thing that all the prototypes had in common is that they were all much too big to be attached to any wind controllers or the like. This problem prevented the author from organizing a study to test the effectiveness in a musical application. Miniaturization of the prototype is certainly possible. Using surface mount parts to replace the breadboards would make the biggest difference. Finally, as was mentioned at the beginning, one of the important goals of this project was to create an input device that can be operated without having to actually look at it during input. To use a typing analogy, the author wanted to give the user the ability to "touch type" regardless of the application. This is potentially a very useful feature in music because the thumb position on many woodwind instruments, for example, is not visible to the eye. You would never see a large cluster of thumb actuated buttons if the user could not see them. With no visual feedback this would be impractical. This input device is then a possible solution to the problem where lots of different choices are desirable but no visual feedback is available.

4.2 Musical Applications

As mentioned in the Background, the ability to make dozens of different discrete selections can be very useful in increasing expressiveness during a live performance. In some instruments, like wind controllers, the thumb plays a central role in changing parameters, because it is not directly in playing notes, which are done by the fingers. In this section we will describe the most relevant musical instruments mentioned in the Background and discuss the effectiveness of their thumb control scheme in light of the final prototype.

4.2.1 Example of a New Performance Technique for Gestural Controllers

In commercial wind controllers like Yamaha's WX5 [25], and Akai [1], changing program numbers, or effects can be quite a complicated procedure that requires the musician to actually stop playing the instrument in order to make the desired change. Assuming that the final prototype can be miniaturized enough to fit on a wind controller, it could provide an improvement on Yamaha's or Akai's technique. The following will describe a possible solution using the final prototype.

As was already described in great detail in the final prototype has two touch sensitive strips and a switch. In this particular musical application, actuating the switch toggles between two modes: Program Change Mode, and Effects Mode. Two LEDs, red and green, could be placed in a visible position on the instrument to notify the musician what mode the instrument is currently in.

Making Program changes only involves one step and will be explained first. Each of the 27 different choices is mapped to a program number. In Program Change Mode, the musician would simply make the desired choice and the program number would change immediately.

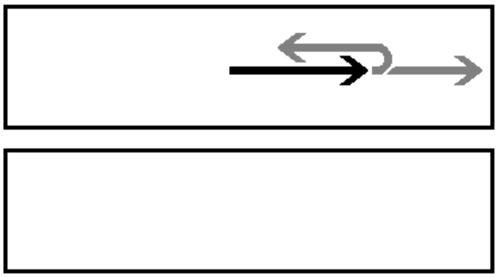

Fig. 4.1 New performance technique for one-thumb input.

The Effects Mode is different than the Program Change Mode because giving the musician the ability to change effects parameters in real-time during a performance is preferred. Each of the 27 different choices are mapped to effects. The procedure would look as follows: 1) The musician selects one of the 27 choices in Effects Mode. In other words, there could be up to 27 different effects mapped to the choices. 2) The choice would quickly be recognized by the input device. We will designate this as the “end of the stroke”. Instead of lifting the thumb off the input device, the musician’s thumb would remain in contact with the strip(s). 3) The thumb now controls a predetermined parameter of the effect selected in the previous step. In this implementation it does not matter what strip is being touched. Only the relative location of the thumb with respect to the “end of the stroke” is what matters. If the thumb remains in the “end of the stroke” position, then the parameter will not change, but remain the same. If however, the thumb moves up the slider(s), then the parameter will increase. Similarly, if the thumb moves down from the “end of the stroke”, then the parameter will decrease. These changes are relative to the previous value of the parameter. Therefore, the strips would have to be considerably longer than the longest stroke, perhaps twice as long, in order to give the musician room to control the effect parameter in either direction at the “end of the stroke”. For example figure 4.1 shows an example of this. The black arrow represents the same choice as in the third row of the first column in figure 3.18 which is mapped to a particular effect. The tip of the black arrow is the “end of the stroke” meaning the choice was recognized and the parameter of the effect mapped to that choice can now be increased or decreased. We will assume that moving to the right increments the parameter, and moving to the left decrements it. The grey arrows represent these two possibilities. If the thumb would continue along the path of the grey arrow pointing right, then the parameter would increase. Moving the thumb along the path of the grey arrow pointing left would decrease the parameter. The amount that the parameter is increased or decreased depends on the length of the stroke. As one can see, the strips had to be lengthened relative to allow for the thumb to move in either direction without running out of room.

Just like in the other figures, the tail represents the starting position when the thumb first makes contact. Once the stroke is recognized the effect that is mapped to this stroke is selected, and the musician can now move up or down, represented by the dotted line, from “end of the stroke” position to increase or decrease the effect parameter respectively.

4.2.2 Application: The Pipe

As was described in the Background chapter, The Pipe simply uses a rotary potentiometer that scrolls through MIDI program change numbers to make voice changes. One problem with this technique is that it you cannot “jump” to a desired number. Instead, you would likely have to scroll through several program numbers before reaching the desired one. Another problem is that the rotary potentiometer does not give any tactile feedback, so the musician would not know what voice has been selected unless it is actually played. Even with some kind of visual feedback, for example displaying the current program number on a computer screen, the musician would still need to take great care to make sure he/she does not overshoot the desired program number. This could be quite distracting if the musician is in the middle of a difficult musical passage. Scavone [20] acknowledges that it was very inconvenient to use the potentiometers and suggested that using a scrolling wheel would be more ergonomic. However, this would not resolve the lack of both visual and a scroll wheel with indents would not add enough in terms of tactile feedback. The technique mentioned in the previous section would solve both the visual and tactile feedback problem, and give the musician the ability to “jump” to any program change desired with a single stroke of the finger, which would be both faster and require less concentration and effort.

4.2.3 Application: Bento-Box

Originally, the Bento-Box used thumb actuated sliders for the thumb to control the key, scale, and chords. In each case, the technique would involve moving the slider to the desired position and then activating this new setting by pushing a button. In other words, the travel of the slide potentiometer would be subdivided into different sections, each of which would be mapped to a particular setting. For example, if the current scale setting was the major scale, then changing it to a minor scale would require the musician move the slider to the minor scale position and then actuate the button. Only after actuating the button would the change become the current setting. This technique also has several problems. In theory, the musician could prepare for the scale change by moving the slider into the desired position well before it is needed and then pushing the button to make that selection the current one. At first glance, this does not seem like something a musician would want to do every time the melody changes. Furthermore, putting the slide potentiometer in the right position while playing, would probably be too much of a distraction. In fact, Hatanaka [9] abandons the idea completely and decides to use the sliders to change effects parameters instead. But using slide potentiometers does not seem like an ideal choice for this kind of application either. For one, the thumb does not move in a straight line, and slide potentiometers might have too much friction. The technique for controlling effects parameters described in the previous section might offer a solution. The strips of the prototype could be curved like some of Hatanaka [9]'s more ergonomic early design concepts.

4.2.4 Application: Yamaha and Akai Wind Controllers

Both the Yamaha [25] and Akai [1] have similar solutions for making program changes. Unfortunately, in either case the musician needs to stop playing musical notes and change the mode of the instrument in order to make the desired program number changes. In the case of the WX5 [25], the musician would first need to actuate a special key with the right thumb to change the mode from playing to program change, and then use two of the note playing keys controlled by the fingers of the left hand to increment or decrement the program number. Another technique would involve using the playing keys for numeric entry. Ten of the note playing keys would be mapped to the digits from 0 to 9, and the program number would be entered directly. The Akai [1] technique for incrementing/decrementing the program numbers is almost identical except it does not use the playing keys. However, the second Akai [1] technique uses the playing keys, but it does not use them for numeric entry. Instead, each key is mapped to a particular predetermined program change number. Unlike the Yamaha [25] model, the Akai [1] can control effect parameters right on the instrument. Again, the musician cannot play the instrument when making effects changes. The procedure is very complicated and involves pressing the two buttons then, depending on which effect is desired, the musician needs to go through a procedure of releasing one of the buttons, pressing another, waiting for the desired effect to be displayed and then releasing all the buttons. Only then can the effect parameter be incremented or decremented, which clearly can take several seconds if not tens of seconds. Using the prototype technique described above would make it possible to change program numbers or effects parameters with a single stroke of the thumb. This would probably be a much better solution making it both easier and faster to make these types of changes during a live performance.

4.3 Text Entry Applications

As mobile computers have gotten progressively smaller and more powerful, a suitable interface for word processing has become necessary. In this section the best prototype's application in text entry will be discussed. The best prototype will be compared with other common text entry methods currently in use such as the cell phone keypad and the miniature qwerty keyboard. The shortcomings of these commercially available input methods will be described and whether or not the features of the best prototype can possibly solve these problems.

Table 4.1 will be used to focus the discussion on the important features of each input device. The first column describes the input method and the second column labeled 'Footprint' describes how many times bigger the different input devices are compared to the cell phone keypad which is the smallest. The third column labeled 'Keys' represents the total number of keys on a typical device. The prototype does not have any keys, but it does have three regions namely left, middle and right, so the value for 'Keys' in this case would be 3. Perhaps the most significant column is the one labeled 'Cluster'. When typing, the digits (fingers and thumbs) divide up the keyboard into groups or clusters. Each digit is responsible for actuating the keys in its own cluster [15]. The numbers in the 'Cluster' column are calculated by taking the number of keys that are most used during regular word processing and dividing it by the number of digits used in typing. The fourth column labeled 'Touch typing' describes whether or not the user can touch type with the input device. Finally, the last row labeled 'One-handed' describes whether or not the user can type comfortably with just one hand. Each input device will be analyzed individually in the next subsections.

Input Method	**Footprint**	**Keys**	**Cluster**	**Touch Type**	**One-handed**
Laptop	20	48	6	Yes	No
Keypad	1	12	6	No	Yes
Mini Qwerty	2	34	17	No	No
Prototype	1	3	3	Yes	Yes

Table 4.1 Text Entry Comparison.

4.3.1 Laptop Keyboard

One of the most common mobile computers is the laptop. The laptop keyboard layout is practically identical to the computer qwerty keyboard. This gives laptop users the ability to touch type just as fast and efficiently on their laptops as they do on their computers. A 'Footprint' of 20 times the size of a cell phone keypad is a rough estimate. If we take the area below the keys, the Footprint could be closer to 30. There are usually close to 80 keys on a laptop keyboard. However, it was determined that there are approximately 48 keys used in text editing: 36 alphanumeric characters, and twelve others including punctuation, Shift, Enter, and the arrow the keys. The cluster value of 6 was arrived at by dividing the 48 keys by 8 fingers since both thumbs are normally only used to press the space bar, and were not used in the calculation. The low cluster value is the reason touch typing is so fast and efficient on a laptop keyboard. And if we only took the alphanumeric characters into account the cluster value would be even lower. In practice, the index fingers control 8 keys each, while the other fingers control 4 each. An experienced touch typist has no trouble locating and hitting a particular key in a particular cluster. This makes touch typing on a laptop keyboard fast and efficient [15]. However, the laptop's large size and weight do not make it a convenient solution for mobile computing.

4.3.2 Cell Phone Keypad

Cell phones are by far the most common handheld devices today. Unlike the laptop, the cell phone is small enough to carry on one's person. The relatively small number of keys means that the Footprint is one of the smallest in Table 4.1. The cell phone keypad was designed to be used with a single thumb. However, two thumbs are normally used in text entry. Although the cluster value is the same as the laptop, the small keys and the fact that three or four letters are grouped on each key makes it practically impossible to touch type. The keypad is really only suitable for sending very short text messages [5] [14] [7].

4.3.3 Mini Qwerty Keyboard

A third mobile device, the handheld computer, attempts to bridge the gap between cell phone and laptop. A miniature version of the qwerty keyboard is crammed onto the hand held (figure 2.10). Typing is done by both thumbs while gripping the device. The space available on a handheld computer is very limited. A little less than half the space is usually

Fig. 4.2 The iPhone QWERTY software keyboard.

taken up by the keyboard, which results in a smaller display size (figure 2.10). Other related devices, like the iPhone (figure 4.2), use a similar approach, but instead of using actual keys, display an image of the keyboard on a touch screen. Although these keyboards, known as soft keyboards, do not have the tactile feedback of physical keys, they can increase the display size by disappearing when they are no longer necessary. Also, the users could have the option of picking which soft keyboard configuration or layout they prefer, or toggling between different sets of keys, which is obviously not possible with a real keyboard [14]. The cluster value was calculated by dividing the number of keys on a typical miniature qwerty keyboard, namely 34, by 2 thumbs. In other words, the keyboard is divided into a left side and a right side. Each thumb is responsible for 17 keys in its cluster. The very high cluster value means that each thumb has to locate and actuate one small target key among 17 keys in the cluster. This makes regular typing more difficult and touch typing impossible.

4.3.4 The Final Prototype

All the text entry methods described above make an assumption that the keypad is the de facto standard for any mobile text entry input device. Buttons are arguably the most important innovation in human-machine interfaces. Nonetheless, buttons need to be a certain size in order to be usable by the general public and there is a limited number that can be crammed onto a cell phone. The prototype in this paper suggests that there is potential for exploring interfaces for the thumb other than the keypad. Potentially having 54 different choices (figure 3.18) is more than enough for text editing as they would include all the alphanumeric characters, punctuation, and special keys, like Tab for example, used when writing documents. Of course, the assumption is that the prototype could be miniaturized enough to fit on a cell phone, but it is not hard to imagine replacing the keypad with two touch sensitive strips. Relying on small keys for mobile text entry does not look like it could ever make touch typing possible. The concept of this prototype might be a more likely avenue for developing a suitable touch typing input device for handheld devices.

Chapter 5

Conclusions and Future Work

This chapter summarizes the work presented in this thesis, offers some conclusions, and discusses possible design directions this research will take in the future

5.1 Conclusions

A one-thumb input device which has a very small footprint and allows the user to make dozens of choices without the need to look at the input device does not exist in commercial products. However, such a device would be very useful in many applications including mobile computing and music. The goal of this thesis was to explore new designs that could achieve this.

Several of the current techniques for thumb control in music instruments were surveyed. In most cases, the thumb was simply used to actuate an FSR in order to control certain continuous musical parameters like volume or timbre. However, in some cases the thumb was used for more sophisticated purposes like making program number changes or selecting and editing effect parameters. The most popular thumb technique for entering text on a mobile device was also described.

All these different musical and typing techniques were found to be inadequate in some respect and the author set out to create a novel input device. The first step in the design process was analyzing the movements of the thumb and categorizing them. There were three types of thumb movements observed namely: pushing states, touching states, and lateral stroke states. The combination of these different states served as the inspiration for the various designs. The first and second design used the idea of using combinations of pushing states and touching states and comprised a joystick and switches. The design was flawed because it turned out that pushing states and touching states were not as independent as was assumed. On the other hand, touching states and lateral stroke states were shown to be independent. The third, fourth and fifth prototypes were based on this concept and all used two touch sensitive strips in their design. However, the touch sensitive strips used in the the third and fourth prototypes were inadequate because they required substantial downward pressure to properly track the thumb's movement making the prototypes too difficult to use. This problem was solved in the fifth and final prototype by using capacitive sensors instead of pressure sensors for the strips.

In the discussion, one-thumb input techniques used in wind controllers or for text entry were analyzed. The current text entry techniques were assessed as either too tedious, like T9 and multitap, or requiring too much concentration, like the crammed miniature qwerty keyboard. In musical gestural controllers, the sensors actuated by the thumb were usually buttons or potentiometers. The thumb was used to control a variety of parameters including: making program number changes, changing the key, controlling effects, to name a few. However, the solution was not ideal in any of these cases because either the musician had to stop playing notes to make the changes, or making the changes was to distracting or difficult. The final prototype was presented as a possible solution in wind controllers and text entry on mobile devices.

This project has given the author much more insight into input devices and has shown that trying to come up with a one-thumb input technique that is based on buttons might be a flawed approach. It is difficult to imagine exactly how one would be able to touch type with dozens of buttons and only one thumb. This thesis suggests that it might be a good idea to step back and rethink the one-thumb design paradigm.

5.2 Future Work

Although the final prototype worked well, there are several major improvements necessary to truly be able to evaluate it's effectiveness as a one-thumb input device. For one, the final prototype was far too large. Using surface mount components on a custom printed circuit boards would decrease the size of the electronics significantly. Printed circuit boards could also be used for the actual touch sensitive strips. In fact, if multi-layer boards are employed, the electronic components and the strips could be put on a single board.

The strips would also need better resolution. Only three capacitive sensors were used per strip. Enough resolution for this application could be achieved with five sensors per strip. This would give ample resolution to distinguish between left, middle, and right thumb positions.

Finally, the input device would need to be properly evaluated for both music and text entry applications. In the case of text entry, a usability study comparing the prototype to cell phones and blackberry type devices could be done. In the case of music, the input device could be attached to a wind controller and musicians could be surveyed on their opinion of the input device.

References

[1] Akai. *EWI4000s Electric Wind Instrument*. Akai.

[2] M. Burtner. Noisegate 67 for metasaxophone. In *Proceedings of the New Interfaces for Musical Expression*, 2002.

[3] P. Cook. Principles for designing computer music controllers. In *Proceedings of the New Interfaces for Musical Expression*, 2001.

[4] S. de Laubier and V. Goudard. Meta-instrument 3: A look over 17 years of practice. In *Proceedings of the New Interfaces for Musical Expression*, 2006.

[5] M. D. Dunlop and A. Crossan. Dictionary based text entry method for mobile phones. In *Proceedings of Second Workshop on Human Computer Interaction with Mobile Devices*, 1999.

[6] A. Freed and F. M. Uitti. Augmenting the cello. In *Proceedings of the New Interfaces for Musical Expression*, 2006.

[7] J. P. Gong and P. Tarasewich. Alphabetically constrained keypad designs for text entry on mobile devices. In *Proceedings of the Conference on Human Factors in Computing Systems*, 2005.

[8] H. Gutowitz. Barriers to adoption of dictionary-based text-entry methods: A field study. Technical report, Eatoni Ergonomics, 2003.

[9] M. Hatanaka. Ergonomic design of a portable musical instrument. In *Proceedings of the New Interfaces for Musical Expression*, 2003.

[10] *High-Precision MicroJoystick Integration Guide*. Interlink Electronics.

[11] A. Kapur, A. J. Lazier, P. Davidson, R. S. Wilson, and P. R. Cook. The electronic sitar controller. In *Proceedings of the New Interfaces for Musical Expression*, 2004.

[12] H. Kober, E. Skepner, T. Jones, H. Gutowitz, and S. Mackenzie. Linguistically optimized text entry on a mobile phone. Technical report, Eatoni Ergonomics, 2001.

[13] D. Lebel and J. Malloch. The g-spring controller. In *Proceedings of the New Interfaces for Musical Expression*, 2006.

[14] I. S. MacKenzie and R. W. Soukoreff. Text entry for mobile computing: Models and methods, theory and practice. *Human-Computer Interaction*, 17:147–198, 2002.

[15] R. McRoberts. *Keyboard Skills*, chapter 2, page 10. Curriculum Corporation, 2003.

[16] E. R. Miranda and M. M. Wanderley. *New Digital Musical Instruments: Control and Interaction Beyond the Keyboard*, chapter 1, page 2. A-R Editions, 2006.

[17] A. Oniszczak and S. Mackenzie. A comparison of two input methods for keypads on mobile devices. In *Proceedings of the Nordic Conference on Human-Computer Interaction*, 2004.

[18] A. Pavlovych and W. Stuerzlinger. Less-tap: A fast and easy-to-learn text input technique for phones. *Graphics Interface*, pages 97–104, 2003.

[19] *Capacitance to Analog Converter*. Quantum Research.

[20] G. P. Scavone. The pipe: Explorations with breath control. In *Proceedings of the New Interfaces for Musical Expression*, 2003.

[21] S. Schiesser and C. Traube. On making and playing an electronically-augmented saxophone. In *Proceedings of the New Interfaces for Musical Expression*, 2006.

[22] E. Sinyor and M. M. Wanderley. Gyrotyre: a dynamic hand-held computer-music controller based on a spinning wheel. In *Proceedings of the New Interfaces for Musical Expression*, 2005.

[23] M. Waisvisz. The hands, a set of remote midi-controllers. In *Proceedings of the International Computer Music Conference*, 1985.

[24] D. Wigdor and R. Balakrishnan. A comparison of consecutive and concurrent input text entry techniques for mobile phones. In *Proceedings of the Conference on Human Factors in Computing Systems*, 2004.

[25] Yamaha. *Yamaha WX5 Wind MIDI Controller Owners Manual*. Yamaha.