Pressure Widgets

Gonzalo Ramos, Matthew Boulos, Ravin Balakrishnan
Department of Computer Science
University of Toronto
bonzo | mboulos | ravin @dgp.toronto.edu
www.dgp.toronto.edu

ABSTRACT

Current user interface widgets typically assume that the input device can only provide x-y position and binary button press information. Other inputs such as the continuous pressure data provided by styluses on tablets are rarely used. We explore the design space of using the continuous pressure sensing capabilities of styluses to operate multi-state widgets. We present the results of a controlled experiment that investigates human ability to perform discrete target selection tasks by varying a stylus' pressure, with full or partial visual feedback. The experiment also considers different techniques for confirming selection once the target is acquired. Based on the experimental results, we discuss implications for the design of pressure sensitive widgets. A taxonomy of pressure widgets is presented, along with a set of initial concept sketches of various pressure widget designs.

Categories and Subject Descriptors: H.1.2 [**Models and Principles**]: User/Machine Systems – *Human Factors; Human Information Processing*; H.5.2 [**Information Interfaces and Presentation**]: User Interfaces -*, Interaction Styles, Input devices and strategies, Theory and methods*.

General Terms: Design, Experimentation, Human Factors

Keywords: Pressure input, pen-based interfaces, isometric input, pressure widgets

INTRODUCTION

Traditional WIMP interfaces are designed to be operated by pointing devices with two degrees-of-freedom that map to the x-y position of the cursor, and binary buttons that enable discrete selection. In addition, current GUIs also support concurrent scrolling using the data provided by the scroll wheel or joystick found on most contemporary mice. Additional degrees-of-freedom provided by some input devices are typically not well utilized by these traditional interfaces. In particular, the stylus used with current tablet computers provides an additional continuous degree-of-freedom that is controlled by the amount of pressure applied by the stylus tip on the tablet surface. To date, this pressure input has typically only been used by a few drawing and image manipulation programs, like Adobe Photoshop, to modulate parameters of the active brush, such as stroke thickness or colour opacity.

As tablet computers become more prevalent, it would seem advantageous to more fully utilize this pressure sensing capability of styluses throughout the interface. In addition to the usual x-y positional cursor control and button clicks that the stylus is currently used for, one can imagine using the stylus' pressure to operate widgets that have several discrete states, or to control a continuous variable. This additional input modality could serve to increase the human-computer communication bandwidth, particularly when tablets are used as pure slates with no keyboard.

To increase the use of stylus pressure, appropriate widgets need to be designed. These designs will, in turn, need to be guided by a thorough understanding of the user's ability to control pressure using a stylus. Questions that need to be answered include: between how many discrete levels of pressure can a user easily discriminate; what is the impact of visual feedback and what form should it take; can users with sufficient practice apply different levels of pressure without any visual feedback; and what mechanisms can be used to indicate completion when pressure is used to acquire one of a discrete set of targets in a widget?

In this paper, we first review the relevant literature. We then present a controlled experiment that investigates users' ability to perform discrete target selection tasks by varying a stylus' pressure, with full or partial visual feedback. The experiment also considers different techniques for confirming selection once the target is acquired. In light of the experimental results, we discuss implications for the design of pressure sensitive widgets. We also introduce a taxonomy of pressure sensitive widgets, along with initial concept sketches of several possible widget designs.

PREVIOUS WORK

Amongst the earlier investigations of the use of pressure in user interfaces is work by Herot and Weinzapfel [7]. They explored the ability of the human finger to apply pressure and torque to a computer screen. They implemented and informally tested five interaction techniques that allowed users to control a cursor's position and speed, as well as to push, pull, disperse, and rotate objects on the screen. In their conclusions, benefits of direct manipulation techniques for the learning process are highlighted. They observe that accuracy in such input methods is achievable if continuous, real-time, visual feedback is provided.

Buxton et al. [4] explored touch-sensitive tablet input and suggested that even though pressure control can be difficult in the absence of button clicks or other similar tactile feedback, it still remains a “ripe area of research”. Furthermore, they present an example of a painting application that employs continuous pressure sensing to control the width of the tool being used. This is perhaps the most commonly used application of pressure input today, and is found in applications such as Adobe Photoshop.

Zhai [15] conducted a series of experiments quantifying the effects of varying various dimensions of 6-dof input devices on 6-dof manipulation and tracking tasks. Of his many observations, the most relevant to our present work was the finding that isotonic devices perform best when used for position or zero-order control, while isometric devices are best suited for rate or first-order control.

Srinivasan and Chen [13] conducted a controlled study in which users were asked to follow different time-profiles of forces (constant, sinusoid, and linear ramps) that were displayed on a computer screen. Participants were asked to control the force applied to a pressure sensor using their index finger pad, under a number of different experimental conditions (normal vs. anesthetized fingertip). The experiment sought to measure human ability to control contact force against a rigid object, while determining the impact of different sensory feedback (presence vs. absence of visual feedback). While their results shed some light on human performance for that particular task, their conclusions cannot be easily extrapolated to produce significant design recommendations about the number of pressure levels a human can reasonably discriminate, the learning effects that may occur, or the impact of different types of visual feedback.

Raisamo [10] evaluated one direct manipulation and four pressure-based area selection techniques for an information kiosk with a pressure-sensitive screen. In the study, users were asked to change the radius of a selection circle by changing the amount of pressure applied to the screen. Different transfer functions that mapped pressure to the circle’s radius were used for three of the selection techniques, while the remaining one incrementally increased the circle’s radius based on a pressure threshold. The study reports that even though users had difficulty controlling two of the pressure-based methods, the slowest pressure-selection technique was ranked almost as highly as the direct manipulation technique. Their overall results indicate that appropriately designed pressure-sensitive interaction techniques could be a practical alternative to standard movement-based methods.

Zeleznik et al. [14] present an alternative to the commonly found binary switches on mice. With their *Pop-Through* buttons users press the button lightly to activate its first state (click), and harder to activate its second state (pop). This extra state enables a novel set of interactions.

Within the framework of their GeoZui3D visualization system, Komerska et al. [8] developed a haptic widget, which controls the viewpoint of a large 3D data space. Users interact with this widget using a Phantom haptic feedback device. The principles employed in designing this haptic widget emphasize visual and haptic feedback in order to provide users not only with the current state of the input device, but also with indicators that suggest what possible interactions are available.

Ramos and Balakrishnan [11] introduced pressure sensitive widgets that were designed to provide users of pen-based interfaces not only with a visual indication of the amount of pressure being applied, but also with meaningful feedback intended to make users aware of the consequences of varying the pen’s pressure. Discrete pressure widgets triggered an action once a certain pressure threshold is exceeded, while continuous pressure widgets mapped pressure to the control of a continuous parameter. These widgets were used to control various tasks such as changing the binary state of an element, and controlling the magnification parameter of a given visualization. However, they did not evaluate these widgets in any formal way.

There are currently many commercial isometric input devices that sense and utilize pressure information in some meaningful way. Examples include IBM’s Trackpoint joystick, which enables 2D scrolling to be performed independently of the x-y position of the mouse; the DualShock2 controller for the Sony PS2 gaming console, whose buttons translate the pressure the user applies to actions in a game; and of course styluses on digitizing tablets such as the Wacom Intuos, which have typically been used by artists to vary brush characteristics in drawing and painting programs. There are also exciting emerging technologies that are capable of sensing, to some degree, the pressure that a user applies to them. Examples of these are Rekimoto’s SmartSkin [12], and Mitsubishi’s Diamond-Touch table [5].

In summary, our review indicates that while there is a rich literature on the use of pressure at the user interface, there has not been a systematic investigation into human ability to control pressure sensitive styluses, nor into the design space of pressure sensitive widgets. Thus, this is an area that is ripe for further research.

EXPERIMENT

Goals

The objective of this study is to investigate human ability to perform discrete selection tasks by controlling stylus pressure. This includes determining the number of levels of pressure a user can comfortably discriminate between when using a stylus, and the impact of visual feedback. We also compare four techniques for confirming selection after the target is located by applying the required pressure.

Apparatus

We used a Wacom Intuos tablet with a wireless stylus with a pressure sensitive isometric tip that does not provide any distinguishable haptic feedback. The stylus provides 1024 levels of pressure, and has a binary button on its barrel. The tablet's active area was mapped onto the display's visual area in absolute mode.

The experiment was done in full-screen mode, with a black background color, on a Dell UltraSharp 1800FP 18-inch Flat Panel LCD Monitor running at a native resolution of 1280 by 1024 pixels. The experimental software ran on a 2GHz P4 PC with the Windows2000 operating system.

Task and Stimuli

A serial target acquisition and selection task was used. Stylus pressure was used to control the movement of a small blue circle cursor along a vertical line. 1024 pressure values were mapped uniformly to a spatial distance of 256 pixels. A set of consecutive rectangles were drawn along the line's length. The size of the rectangles was experimentally manipulated. During each experimental trial, one of the targets was highlighted in green, and the user's task was to apply the appropriate amount of pressure to move the blue circle cursor into that target. When the cursor enters the target, the target color changes to red.

We use two different visual feedback conditions (Figure 1): *Full Visual* (*FV*) and *Partial Visual* (*PV*). The *FV* condition shows the target in context with the other adjacent ones and provides continuous feedback in the form of the cursor's position along a vertical line. In the *PV* condition, only the target is visible, and the cursor is only shown at the start of the trial. Once movement begins, the cursor is hidden, and the user has to rely on proprioceptive cues and memory to accurately determine the amount of pressure to apply in order to get the hidden cursor into the target. This simulates the condition where expert users may be able to use pressure for quick selection in an eyes-free manner, similar to behavior exhibited by expert users of Marking Menus [9]. However, this condition is not completely free of visual feedback: as in the *FV* condition the target color changes from green to red when the hidden cursor is inside the target. In other words, feedback is provided at the final stage of the task. A similar approach has been used successfully in previous experiments studying the limits of kinesthetic cues in interface tasks [1].

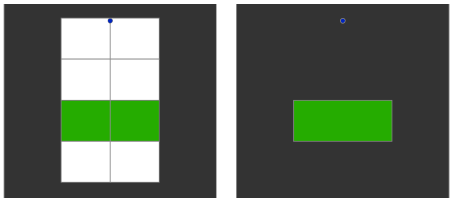

Figure 1: Visual feedback conditions. (left) *FV*: green target is shown in context with other possible targets; the blue cursor is always visible. (right) *PV*: green target is shown in alone, the blue cursor disappears once pressure is applied.

Once the cursor is in the target, there has to be a mechanism for the user to confirm selection. In standard GUI interfaces, this is typically done by clicking the mouse button. An analogous mechanism for the stylus would be clicking the barrel button. In practice, however, the ergonomics of the stylus makes this action less than ideal since users often rotate the stylus and the button may not always be in a position to facilitate clicking. Further, clicking the barrel button can cause inadvertent movement in the x-y direction, which is problematic if we are to design interaction widgets where pressure is used in conjunction with x-y stylus movement. In contrast, mouse buttons are orthogonal to x-y movement, reducing the possibility of inadvertent movement during button clicks.

We therefore investigate various alternatives. A total of four techniques were tested: *Click:* pressing the stylus' barrel button; *Dwell:* maintaining the cursor within the target for a prescribed amount of time (in our experiment, a 1 second delay was used); *Quick Release*: quickly lifting the stylus from the tablet's surface; and *Stroke:* quickly make a spatial movement to the right. Each method has a particular signature in terms of pressure, spatial position, and button state, as illustrated in Figure 2.

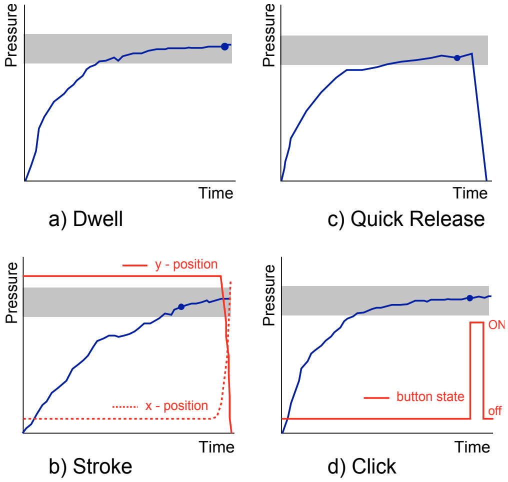

Figure 2: Signatures for the selection methods. The grey band shows target pressure range. The large dot on each curve marks where the selected pressure value is taken.

Participants

Seven female and five male volunteers, 18-34 years old, participated in the experiment. All were right-handed and had little to no prior experience using pressure sensitive devices such as the stylus used in the experiment.

Procedure and Design

A within-subjects full factorial design with repeated measures was used. The independent variables were selection method (Click, Dwell, Quick Release, and Stroke), visual feedback condition (*FV*, *PV*), the distance from the starting point to the target (*D*= 205, 410, 615, 820) and the target's width (*W*= 85, 102, 128, 170, 256). Distance and width are expressed in pressure units.

Recall that we used a fixed mapping of pressure values to cursor movement (1024 pressure values were linearly mapped to 256 pixels of cursor movement). As such, changing *W* changes the number of divisions of the 256 pixel potential target space. We used this approach rather than variable pressure to spatial mappings in order to resemble the designs we anticipate for pressure widgets where the overall widget size will likely remain constant with appropriate subdivisions into selectable targets, much like Marking Menus retains a uniform size regardless of number of menu items [9]. Furthermore, keeping the pressure to spatial movement mapping constant will likely facilitate user's ability to develop haptic memory of various pressure levels.

In order to keep the experiment balanced, special care was taken when choosing the *W* and *D* values such that targets were appropriately distributed throughout the potential target space (Figure 3). This choice, however, resulted in *D* not always defining the distance from the start to the centre of the targets, but instead to *some* location in the targets. Because we are also interested to see if this pressure-controlled target acquisition task obeys Fitts' law [6], we compute the index of difficulty (*ID*) for each condition using the amplitude *A* (i.e., the distance to the centre of the relevant target) rather than *D* (Figure 3).

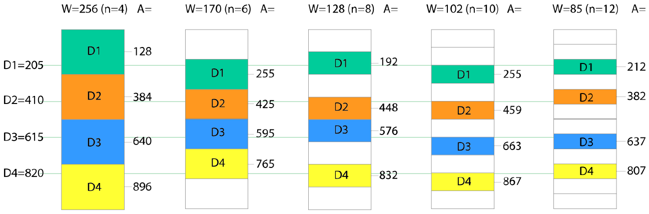

Figure 3: Pressure space for the study. *W* dictates the number of pressure levels, *n*; *D* defines a target for a given *W*; *A* is the *amplitude* used to compute Fitt's *ID*

Participants were randomly assigned to 4 groups of 3 participants each. In each group, participants were exposed to all four selection methods, whose order of appearance was balanced using a Latin square. For each selection method, participants were asked to complete two sessions of 5 blocks each. In the first session the FV feedback was used, and in the second session the PV feedback was used. Each block consisted of trials for all 20 D-W conditions, repeated 3 times, which resulted in 60 trials per block. Presentation of trials within a block was randomized. In summary, the experiment consisted of:

12 participants x
4 selection methods x
2 visual feedback conditions x
5 blocks x
20 *D-W* conditions x
3 repetitions
= 28800 target selection trials.

Prior to performing trials for each selection method, participants were given a short warm-up set of trials to familiarize themselves with the selection technique. Participants were instructed to perform the task as quickly and accurately as possible. Participants could take breaks between trials, and breaks were enforced between changes of visual feedback condition. The experiment lasted approximately 2 hours for each participant. A short questionnaire was administered at the end of the experiment to gather subjective opinions.

For each trial, we collected all the stylus data events (position, pressure, and time). This allowed us to measure the time it took to perform a task, the result of the task (i.e. success or failure) as well as any extra information such as the number of times the cursor enters and leaves a target before the participant selects it, or changes in the stylus' spatial position. An audible beep provided error feedback if a selection was made outside the target. Timing began the moment the stylus came into contact with the tablet's surface (i.e. the tablet reported a pressure > 0) and ended when the appropriate selection technique was executed.

Performance Measures

The dependent variables were *movement time MT*– defined as the time from when the stylus came into contract with the tablet's surface until the appropriate selection technique was executed; *error rate ER* – defined as the percentage of trials for a particular condition that resulted in erroneous selections; and *number of crossings NC* – defined as the number of times the cursor enters or leaves a target for a particular trial, minus 1 (e.g., *NC*= 2 for a task where the user overshoots and reacquires the target). These measures complement each other: while *MT and ER* give us an indication of the overall success rate, *NC* tells us about the degree of pressure control that participants exerted.

Results

We removed outliers from the data set. A trial was considered an outlier if the time it took to complete the task was beyond 2 standard deviations from the mean task completion time. A total of 1326 outliers were discarded, representing 4.6% of the data collected

Selection Techniques

Analysis of variance showed a significant main effect for selection technique on *MT* for the *FV* ($F_{3,6}$=790.96, *p*<.0001) and *PV* ($F_{3,6}$=729.94, *p*<.0001) conditions. Pairwise means comparisons showed significant difference between all pairs of techniques (*p*<.0001). Overall, the fastest selection technique was *Quick Release*, followed in order by *Click*, *Stroke*, and *Dwell* (Figure 4.a).

Error rate *ER* was also significantly different across selection methods for both *FV* ($F_{3,6}$=118.36, *p*<.0001) and *PV* ($F_{3,6}$=279.47, p<.0001) conditions (Figure 4.b). Pairwise means comparisons showed significant difference between all pairs of techniques (*p*<.0001) except *Stroke* x *Click* (*p*= 0.3911). *Dwell* had the lowest error rate followed in order by *Quick Release*, *Stroke*, and *Click.*

As a measure of pressure control, the number of target crossings *NC* was also significantly different across selection methods for both *FV* ($F_{3,6}$=166.53, p<.0001) and *PV* ($F_{3,6}$=164.62, p<.0001) conditions (Figure 4.c). Pairwise means comparisons showed significant difference between all pairs of techniques at p<.0001, with the exception of *Dwell* x *Click* that was significant at p<.05 and *Q.Release* x *Click*, which was not significant (p = 0.0848). With this measure, the relative ordering of selection techniques differed depending on visual feedback. For the *FV* condition, *Stroke* was the most difficult technique to control, followed by *Quick Release*, *Click*, and *Dwell*. For the *PV* condition, *Dwell* was the most difficulty, followed by *Stroke*, *Click*, and *Quick Release*. The overall higher *NC* rate for the *PV* condition is consistent with our observations that without visual feedback, users had to resort to "fishing around" in order to acquire the desired target.

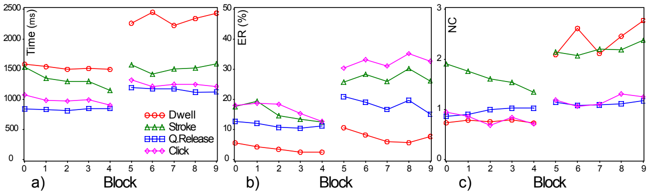

Figure 4: Effect of selection on (a) *MT*. (b) *ER*. (c) *NC*. The left and right half of each graph shows data for the *FV* and *PV* condition, respectively.

For the *FV* condition, significant learning effects with respect to *MT* ($F_{4,44}$= 19.21, p<.001) were observed across blocks, with all selection methods improving except for *Quick Release*. Similarly for *ER* ($F_{4,44}$= 8.28, p<.001). With regards to pressure control, significant improvements were observed in *NC* for the *Click* and *Stroke* techniques, consistently low *NC* was seen for *Dwell*, while *NC* for *Quick Release* degraded slightly over time.

For the *PV* feedback condition, *MT* slightly improved with practice for the *Quick Release* and *Click* techniques, however they did not approach the times seen in the *FV* condition. Time performance for *Dwell* and *Stroke* was erratic, and no trend could be clearly identified. Some improvement in error rate was seen for the *Dwell* and *Quick Release* techniques, but not for *Click* or *Stroke.* Erratic *NC* values were observed for *Dwell*, while the three techniques had high *NC* values that were fairly constant throughout.

A significant selection x block interaction for *MT* ($F_{15,132}$= 160.62, p<.001), *ER* ($F_{15,132}$= 8.28, p<.001), and *NC* ($F_{15,132}$= 2.63, p<.05) in the *FV* condition indicates that participants progressed at varying rates for the different selection techniques.

In the post experiment questionnaire, participants were asked to rate on a 7-point "agree – strongly disagree" Likert scale if they believed they made many mistakes. The average result was 5.1, falling in the "somewhat agree" slot. Participants also ranked the selection techniques according to how easy they were to use. *Quick Release* was ranked highest, followed by *Dwell*, *Click*, and *Stroke.* This resembles the ranking for error rate, which may suggest that participants associate successful selections with ease of use.

Discernable Number of Pressure Levels

One of the main purposes of this study was to determine how many discrete levels of pressure, *nLevels*, users can discriminate between at a decent level of performance. Except for *Stroke*, error rates for the *FV* condition when *nLevels*<= 6 ranged from 1 – 8% and were not significantly different within each selection method, for all selection methods (Figure 5.a). Also, *NC* values for all techniques reach a plateau at *nLevels*= 8, except for *Stroke* which continues to deteriorate (Figure 5.b). Best performance is seen for *nLevels*<= *6*, where *NC*< 1.8 for all but the *Stroke* technique. These results indicate to us that 6 levels of pressure is a reasonable number that can be reliably differentiated with adequate visual feedback.

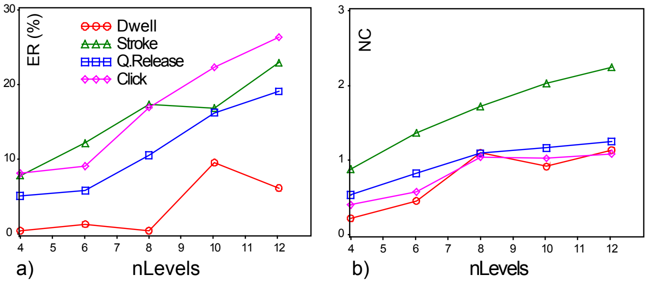

Figure 5: Effect of *nLevels* on *ER* (a) and *NC* (b) (FV condition)

For the *PV* condition, both *ER* and *NC* have increasingly poor levels for *nLevels* > 4. At *nLevels*= 4, *NC* values remain below 1.78 for all selection techniques, and *ER* values remain close to 5% for *Dwell* and *Click*; and 15% for *Quick Release* and *Stroke*.

Effect of Visual Feedback

We included a partial visual feedback (*PV*) condition in our experiment to simulate the situation where expert users would perform pressure based selection without looking at the visual feedback, relying instead on their haptic memory of the amount of pressure to apply. In particular, all participants performed the experiment with full visual feedback (*FV*) first, in order to gain expertise with the techniques, before attempting expert behaviour. Overall, the results indicate (Figure 4) that performance in the *PV* conditions are significantly lower than in the *FV* conditions for *MT* ($F_{1,11}$= 1144.51 p<.0001), *ER* ($F_{1,11}$= 470, p<.0001), and *NC* ($F_{1,11}$= 581.99, p<.0001). Given that our experiment lasted approximately 1 hour per participant for the *FV* condition, it is quite likely that this was not sufficient time to develop expert behaviour. However, we cannot rule out the possibility that there will always be a fundamental difference between *FV* and *PV* performance, regardless of the amount of practice.

Conformity with Fitts' Law

Most target acquisition tasks tend to follow Fitts' law, where *MT* is modeled by the following relationship:

$$MT = a + b.\log_2 \underbrace{\left(\frac{D}{W}+1\right)}_{ID}$$

The logarithmic term is called the *index of difficulty* (*ID*) for the target acquisition task. Linear regression of our *MT* data by *ID* for each selection technique (Figure 6.b) indicated high correlations with Fitts' law for *Quick Release* (r^2= 0.9) and *Dwell* (r^2= 0.84), and poorer correlations for *Click* (r^2= 0.74) and *Stroke* (r^2= 0.44). For the *Stroke* technique, this is perhaps unsurprising since it involves two distinctly separate actions: pressure followed by a spatial stroke gesture. The *Click* and *Stroke* technique involved actions that likely affected user's ability to maintain a particular pressure level, resulting in the high error rates observed and thus more variance in the regression.

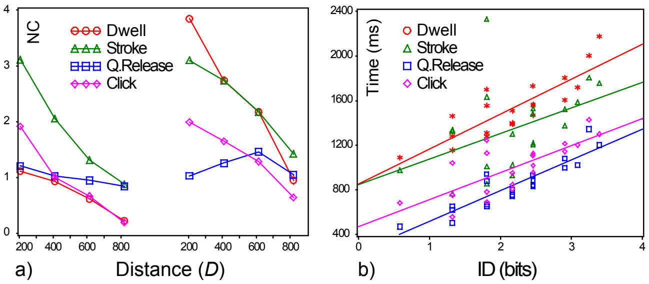

Figure 6: (a) Effect of amplitude on pressure level on *NC*: the left half of the graph shows the *FV* condition and the right the *PV* condition. (b) Linear regression of *MT* data by selection.

Control at Different Pressure Levels

Our observations during the experiment indicated that participants' ability to control pressure varied according to the amount of pressure required. Participants also reported that "the pen was too sensitive" when they tried to acquire a target at a low pressure value. Analysis of variance confirms these observations and shows that *NC* values are significantly different across amplitudes for both the *FV* ($F_{3,33}$= 221.81, p<.0001) and *PV* ($F_{3,33}$= 172.65, p<.0001) conditions. There was also a significant selection method x amplitude interaction for the *FV* ($F_{12,99}$= 72.94, p<.001) and the *PV* ($F_{12,99}$= 61.66, p<.001) conditions. Figure 6.a illustrates these effects.

DISCUSSION

Our results have shown that the different selection techniques have significant effects on the usability of stylus pressure for performing discrete selection tasks. It is important to note that all the techniques have a common first phase: applying the right amount of pressure to move the cursor into the target. It is the second, selection, phase that differs between techniques. Some of the techniques (e.g., *Dwell*) have a smoother transition relationship between the two phases, while others required a distinctly separate action that could have interfered with performance.

The *Dwell* technique was the most accurate and allowed for the highest degree of pressure control (low *NC*), at least in the *FV* condition. This is perhaps unsurprising since the second phase in this technique involves simply waiting for the appropriate time delay to pass without any other movements required. However, this incurs a built in 1 second penalty, resulting in a tradeoff between accuracy, control and time.

Participants consistently rated *Stroke* as being quite difficult. Completion time using *Stroke* depended greatly on people's skill in performing the stroke gesture. Participants learned this gesture at different rates, and performed the selection at different speeds. While some participants performed a quick flick motion, others were very careful and did a slow and controlled motion. Our implementation recognized any significant movement that went to the right as being a valid gesture, but some participants commented that it was more natural for them to perform the gesture in an upward diagonal motion instead of a horizontal one. Difficulties in maintaining a stable pressure value while moving the stylus also contributed to the poor performance of this technique. Our algorithm attempts to compensate for this by estimating the point at which the stroke gesture begins and ignoring pressure fluctuations thereafter. However, this estimation process is not always successful.

Quick Release was subjectively the highest rated, and quantitatively the quickest technique. It was also the most usable in the absence of full visual feedback. This is because the second phase of this technique is very fast, and thus does not prolong the need to maintain a particular pressure level after the first phase.

As might be expected from our earlier discussions about the ergonomics of the pen, we found that the button presses in the *Click* technique interfered significantly with pressure control. Unless the pen's design can be changed significantly, our results indicate that this is not a good technique for pressure based target selection.

Implications for design

The results of our experiment suggest several guidelines for the design of pressure sensitive widgets:

Minimize the interference between movement and selection phases. Movement and selection with an isotonic pointing device (e.g. mouse) occur on uncorrelated (orthogonal) input channels. With an isometric stylus, however, appropriate techniques need to be utilized to minimize inherent interferences. Our *Dwell* and *Quick Release* techniques are good candidates in this regard.

Minimize the interference between spatial x-y movement and pressure channels. A poor visual design may inadvertently make users move the stylus while trying to perform a pressure control task. We observed this phenomenon with our experimental setup. Even though participants were instructed that only pressure had an effect on the target acquisition task, many moved the stylus spatially in the vertical direction, corresponding to the movement direction of the blue cursor (Figure 1). It was interesting to observe that while some users did this inadvertently, others used this motion as an *explicit* way to control stylus pressure. We will elaborate later on widget designs that aim to minimize this source of interference.

Have the "right" number of pressure levels. Our results show that dividing the pressure range into 6 levels or less produces the best performance, while increasing the number of divisions drastically degrades performance.

Provide real-time, continuous feedback. Even though pressure activated target acquisition tasks are achievable without continuous feedback (*PV* condition), pressure control was consistently poor, if not erratic, under this condition for almost all selection methods. While this could improve with lots of practice, it is clear that good feedback is required for pressure sensitive widgets.

Choose a good transfer function. Participants consistently demonstrated less pressure control for low levels of pressure, and described the widget as "very sensitive" at these levels. The simple linear transfer function used in our experiment could be improved to take into account this variation of control at different pressure levels. Investigating what transfer function would be adequate for a particular pressure widget goes beyond the scope of this paper and remains an issue for future research.

PRESSURE WIDGET DESIGNS

Building on our experimental results and our observations from the previous section, we now explore the design space of pressure widgets. To aid our exploration, it is useful to define certain parameters of the design space.

Channel coupling. In our particular case we are interested to what visual attribute of the widget the pressure channel is mapped. In our experiment, the set of rectangular targets (i.e., widget) was fixed in space, while stylus pressure controlled the cursor's position. In other words, the pressure channel was coupled to the cursor's *position*. An alternative approach would be to fix the cursor's position but have the entire widget coupled to pressure and have it move according to variations in stylus pressure. We will say that an input channel can be coupled to: *position* if variation in the channel translates to changes in x-y coordinates; *scale* if it translates to changes in size or scale; and *angle* if it translates to changes in angle or orientation. Although it is possible to consider additional input channels such as x-y position, we will only explore pressure coupling.

Widget visual element. From the last paragraph we saw that pressure can be coupled to a widget's *cursor*. For the purpose of our analysis of the design space we will say that a pressure widget is composed of two distinctive visual elements: *cursor* and *target(s)*. We informally define *cursor* as the visual feature that indicates what item will be chosen if a selection occurs. We will refer to as *targets* the visual representation of the set of items available for selection.

Like similar efforts for input devices by Buxton [3], and for see-through tools by Bier [2], these parameters can be used to create a design taxonomy that describes the nature of possible pressure widgets (Table 1). Assisted by this taxonomy we developed some alternative designs for both discrete and continuous pressure widgets.

	Pressure Coupling		
	Position	***Angle***	***Scale***
Cursor	-Flag (as used in our experiment)	-Rot. Exp. Pie	-Bullseye -Pressure Grid -Pressure MM -TLSlider
Targets	-Flag	-Rot. Exp. Pie	-TLSlider

Table 1: A taxonomy for the design of pressure widgets. Combinations of cells will describe the behaviour for a particular pressure widget.

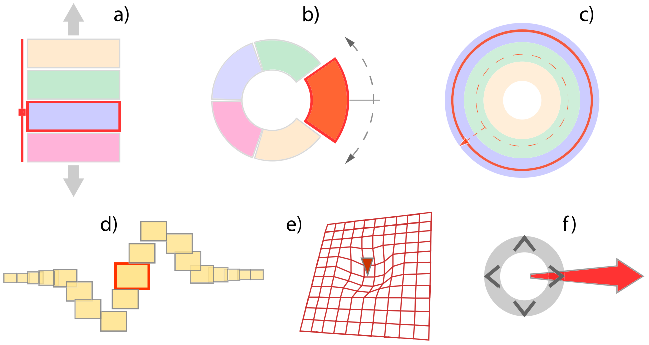

Figure 7: Concept designs for pressure widgets. (a) Flag. (b) Rotating Expanding Pie. (c) Bullseye. (d) TLSlider. (e) Pressure Grid. (f) Pressure Marking Menu

The *Flag* (Figure 7a) couples pressure to *position,* which can be either applied to its *cursor* or *targets.* While our experiment's *Flag* widget coupled pressure to the *cursor*'s position, informal observations indicate that changing the coupling to the *targets*' position (the "whole" widget displaces in response to pressure changes) will alleviate the self-interference resulting from users moving the stylus in x-y space while applying pressure. Since the cursor does not move, the user is less likely to associate cursor movement to requiring stylus movements. The actual performance differences between *cursor* and *target* coupling for this *Flag* widget remains to be studied.

The *Rotating Expanding Pie* (Figure 7b) couples pressure to its *targets*' *angle* and its *cursor*'s *scale*. This widget consists of a set of circular sectors that rotate as the user applies different levels of pressure. The cursor or item currently selected is the sector that intersects an imaginary needle pointing at zero degrees. The targeting stage is also reinforced by an animation that slides and magnifies the sector under the cursor. In this instance we hope to avoid self-interference by means of two factors: a) there is no graphic cursor that the user can see; and b) the spinning, circular nature of the widget may deter users from attempting motion with the pen while targeting.

The *Bullseye* (Figure 7c) couples pressure to its *cursor*'s *scale*. This widget consists of a set of concentric rings that represent the different pressure level intervals. As the user applies pressure with the stylus, a ring cursor expands or contracts to fall into one of the rings. In this case we expect to alleviate self-interference by using a cursor that appears to not move, but “changes size”.

We use the *Twist-Lens-Slider* (*TLSlider*) [11] (Figure 7d) to see if the design matrix could describe existing pressure widget designs. The *TLSlider* is a variation of a fish-eye lens visualization, which displays a linearly ordered set of items and uses pressure to spatially spread items that otherwise would occlude each other. By looking at this behavior the design matrix describes it as having pressure coupled with both the *cursor* and the *targets*’ *scale*.

The *Pressure Grid* (Figure 7e) is a cursor where pressure is coupled with *scale*. It consists of a wire frame or grid that is deformed as the users applies pressure with the stylus. A v-shaped pointer also changes with pressure: the more pressure is applied, the thinner (or pointier) it will become. Preliminary user observations suggest that this is a compelling visualization in which the meaning of varying pressure is very clear, thus reducing interference effects.

The *Pressure Marking Menu* (Figure 7f) taps into the inherent variability in pressure of the *Stroke* selection method and extends the number of items available on regular marking menus [9]. Instead of having one option available at a given path, the pressure marking menu has two or more, depending on the pressure difference between the starting and ending points of the marking gesture. This widget has pressure coupled with the *cursor*’s *scale*.

All these widgets can be slightly modified to take into consideration pressure sensitivity at low levels of pressure. A “dead zone” at the beginning of the pressure range can be mapped onto a small band, a sector, or a ring on the widget, if appropriate. This zone is a place where no item is selected even though pressure is being applied. Once the pressure applied surpasses a certain threshold, the first item will become targeted. Under the same principle, the size of the different items on a widget can vary proportionally to the sensitivity of that pressure zone.

CONCLUSION

We have presented a controlled experiment that investigated human ability to use stylus pressure to perform discrete target acquisition tasks, with different selection techniques. Our results indicate that the *Quick Release* selection technique was preferable overall, and that dividing pressure space into 6 levels is optimal. Appropriate visual feedback was also found to be critical: users were not able to effectively perform eyes-free pressure selection with only an hour of practice. Based on the results of our experiment, we have inferred design recommendations and proposed a design taxonomy of pressure widgets. Initial designs of pressure widgets were also presented.

Future intended work on this subject includes evaluation of these new proposed designs and testing their impact in reducing interference issues, as well as studying and designing appropriate transfer functions for pressure. We will also investigate the effects of using an integrated display/sensing device, such as the Tablet PC and assessing how it differs from our current results and observations.

ACKNOWLEDGMENTS

All who participated in the experiment, members of DGP lab, Microsoft Research for financial support.

REFERENCES

1. Balakrishnan, R., & Hinckley, K. (1999). The role of kinesthetic reference frames in two-handed input performance. *ACM UIST*. p. 171-178.
2. Bier, E., *et al.* (1994). A taxonomy of see-through tools. *ACM CHI Conference.* p. 358-364.
3. Buxton, W. (1983). Lexical and pragmatic considerations of input structures. *Computer Graphics*. 17(1). p. 31-37.
4. Buxton, W., *et al.* (1985). Issues and techniques in touch sensitive tablet input. *ACM SIGGRAPH Conference*. p. 215-224.
5. Dietz, P., & Leigh, D. (2001). DiamondTouch: a multi-user touch technology. *ACM UIST*. p. 219-226.
6. Fitts, P.M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *J. of Exp. Psychology*. 47. p. 381-391.
7. Herot, C., & Weinzapfel, G. (1978). One-point touch input of vector information for computer displays. *ACM SIGGRAPH Conference*. p. 210-216.
8. Komerska, R., *et al.* (2002). Haptic interface for center-of-workspace interaction. *IEEE Virtual Reality - Haptics Symposium*. p. 352-353.
9. Kurtenbach, G., & Buxton, W. (1993). The limits of expert performance using hierarchical marking menus. *ACM CHI Conference.* p. 35-42.
10. Raisamo, R. (1999). Evaluating different touched-based interaction techniques in a public information kiosk. *Conference of the CHI Special Interest Group of the Ergonomics Society of Australia*. p. 169-171.
11. Ramos, G., & Balakrishnan, R. (2003). Fluid interaction techniques for the control and annotation of digital video. *ACM UIST*. p. 105-114.
12. Rekimoto, J. (2002). SmartSkin: an infrastructure for freehand manipulation on interactive surfaces. *ACM CHI Conference*. p. 113-120.
13. Srinivasan, M., & Chen, J. (1993). Human performance in controlling normal forces of contact with rigid objects. *Proc of ASME*. 49. p. 119-125.
14. Zeleznik, R., *et al.* (2001). Pop through mouse button interactions. *ACM UIST*. p. 195-196.
15. Zhai, S. (1995). Human performance in six degree-of-freedom input control. Ph.D. Thesis. University of Toronto.