IMPROVING EXPRESSIVITY IN DESKTOP INTERACTIONS WITH A PRESSURE-AUGMENTED MOUSE

A Thesis Submitted to the
College of Graduate Studies and Research
in Partial Fulfillment of the Requirements
for the Degree of Master of Science
in the Department of Computer Science
University of Saskatchewan
Saskatoon

by

Jared Cechanowicz

©Jared Cechanowicz, November 2008. All rights reserved.

PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Liberties of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science
176 Thorvaldson Building
110 Science Place
University of Saskatchewan
Saskatoon, Saskatchewan
Canada
S7N 5C9

ABSTRACT

Desktop-based Windows, Icons, Menus and Pointers (WIMP) interfaces have changed very little in the last 30 years, and are still limited by a lack of powerful and expressive input devices and interactions. In order to make desktop interactions more expressive and controllable, expressive input mechanisms like pressure input must be made available to desktop users. One way to provide pressure input to these users is through a pressure-augmented computer mouse; however, before pressure-augmented mice can be developed, design information must be provided to mouse developers. The problem we address in this thesis is that there is a lack of ergonomics and performance information for the design of pressure-augmented mice. Our solution was to provide empirical performance and ergonomics information for pressure-augmented mice by performing five experiments. With the results of our experiments we were able to identify the optimal design parameters for pressure-augmented mice and provide a set of recommendations for future pressure-augmented mouse designs.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor Carl Gutwin for his guidance and input throughout my graduate career at the University of Saskatchewan. I also wish to thank Sriram Subramanian for introducing me to the world of HCI research, and Pourang Irani for his research assistance. A big thank you to the faculty and staff at the Department of Computer Science, and particularly to the students and staff in the HCI Lab for their help and their friendship.

This thesis is dedicated to my mother Rosanne, my father Edward, and my brother Evan; the truest friends I will ever know.

CONTENTS

Permission to Use i

Abstract ii

Acknowledgments iii

Contents v

List of Tables ix

List of Figures x

List of Abbreviations xiii

1 Introduction 1
- 1.1 Problem 2
- 1.2 Solution 3
- 1.3 Steps in the Solution 4
- 1.4 Evaluation 6
- 1.5 Contributions 7
- 1.6 Thesis Outline 7

2 Related Work 9
- 2.1 Human Control of Pressure 10
  - 2.1.1 Sensing Pressure 10
  - 2.1.2 Applying Pressure 11
- 2.2 Mouse Augmentation 12
  - 2.2.1 Additional Buttons 12
  - 2.2.2 Force Feedback 14
  - 2.2.3 Additional Degrees-of-Freedom 15
- 2.3 Pressure Interaction 16
  - 2.3.1 Discrete Pressure Interaction 17
  - 2.3.2 Continuous Pressure Interaction 19
- 2.4 Non-Pressure-Based Interaction Techniques 21
- 2.5 Alternative Selection Interactions 23
  - 2.5.1 Pressure-Based Selection 23
  - 2.5.2 Non-Pressure-Based Selection 24
- 2.6 Augmented GUI Interactions 25

3 Motivation: Designing More Expressive Interactions 28
- 3.1 A Framework for Augmented Interaction 28
  - 3.1.1 Interaction: Object + Actions 28
  - 3.1.2 Augmentation 31
  - 3.1.3 Input Mechanisms 32

3.1.4 User Task 36
3.2 Applying the Framework 37
3.3 Relationship to Other Models and Paradigms 38
3.4 Summary 39

4 Design Guidelines for a Pressure-Augmented Mouse 40
4.1 Design of a Uni-Pressure Mouse 42
4.1.1 Number of Levels 42
4.1.2 Sensor Location 42
4.1.3 Number of Sensors 43
4.1.4 Discretization of Raw Pressure Values 44
4.1.5 Pressure Control Mechanism 45
4.1.6 Selection Mechanism 45
4.1.7 Visual Feedback 46
4.2 Uni-Pressure User Study 47
4.2.1 Hardware Configuration 47
4.2.2 Performance Measures 47
4.2.3 Methods 48
4.3 Results 50
4.3.1 Trial Completion Time 51
4.3.2 Crossings and Errors 51
4.3.3 Subjective Ranking 55
4.3.4 Discussion 55
4.4 Design of a Dual-Pressure Mouse 56
4.4.1 Dual-Pressure Control Mechanisms 57
4.5 Dual-Pressure User Study 59
4.5.1 Methods 59
4.6 Results 61
4.6.1 Trial Completion Time 61
4.6.2 Crossings and Errors 62
4.6.3 Discussion 63
4.7 Summary 65

5 Design Guidelines for Pressure Buttons 67
5.1 Design of Selection and Action Techniques 69
5.1.1 Selection Techniques with Pressure Buttons 69
5.1.2 Action Techniques with Pressure Buttons 71
5.1.3 Visual Feedback 72
5.2 Selection User Study 73
5.2.1 Hardware Configuration 73
5.2.2 Performance Measures 74
5.2.3 Methods 74
5.3 Results 77
5.3.1 Completion Time 77
5.3.2 Errors and Subjective Feedback 78
5.3.3 Discussion 78
5.4 Action Invocation User Study 80

5.5 Results 82
5.5.1 Completion Time and Errors 82
5.5.2 Subjective Feedback 82
5.6 Summary 84
5.6.1 Observations on the Design of Pressure Buttons 84
5.6.2 Natural Extensions to Pressure Buttons 86
5.6.3 Design Recommendations 89

6 Augmented Interactions 91
6.1.1 Augmented Scroll Buttons 91
6.1.2 Augmented Scroll Thumb 93
6.1.3 Augmented Object Previews 94
6.1.4 Augmented Back/Forward Buttons 96
6.1.5 Augmented Activations 97
6.1.6 Augmented Dragging 99
6.2 User Study of Augmented Interactions 100
6.2.1 Study Apparatus 100
6.2.2 Participants and Tasks 102
6.2.3 Results 102
6.3 Discussion 104
6.3.1 Prediction of Results Based on the Framework 104
6.3.2 Deployment and Adoption of Augmented Interactions 105
6.4 Summary 106

7 Discussion 107
7.1 Summary of Findings 107
7.1.1 Mouse-Based Pressure Control is Similar to Stylus-Based Pressure Control 107
7.1.2 Dual-Pressure Input Can Be Used to Control 64 Discrete Levels 108
7.1.3 Pressure Input Can Efficiently Replace Double-Clicks, But Has Error Issues 109
7.1.4 Users Prefer More Powerful and Controllable WIMP Interactions 109
7.2 Lessons Learned 110
7.2.1 Pressure Input is Relatively Imprecise 110
7.2.2 Effective Pressure-Sensor Buttons Must Have a Large Enough Footprint 111
7.2.3 Pressure Maps Well to User Interest and Confidence 112
7.2.4 Users Want More Powerful GUI Interactions Regardless of Input Device 113

8 Conclusion 115
8.1 Contributions 115
8.1.1 Primary Contributions 115
8.1.2 Secondary Contributions 116
8.2 Future Work 117

8.2.1 Discretization and Linearization Functions 118
8.2.2 Reducing Errors 118
8.2.3 Prototype Design and Testing of Pressure Button and Augmented Button Mice 119
8.2.4 Quantitative Evaluation of Augmented Interactions 119
8.2.5 Pressure Gestures 120

References **121**

A Qualitative Study Questionnaires and Results **127**
A.1 Consent Form 127
A.2 Pressure Augmented Mouse Experiment 1 TLX and Questionnaire 129
A.3 Pressure Augmented Mouse Experiment 2 TLX and Questionnaire 135
A.4 Pressure Buttons Experiment 1 TLX and Questionnaire 139
A.5 Pressure Buttons Experiment 2 TLX and Questionnaire 142
A.6 Augmented Interactions TLX and Questionnaire 145
A.7 Augmented Interactions Questionnaire Results 151

List of Tables

6.1 Augmented scroll buttons as described in terms of the components of the augmented interactions framework in Chapter Three. 92

6.2 Augmented scroll thumb as described in terms of the components of the augmented interactions framework in Chapter Three. 93

6.3 Augmented object previews as described in terms of the components of the augmented interactions framework in Chapter Three. 95

6.4 Augmented back/forward buttons as described in terms of the components of the augmented interactions framework Chapter Three. 96

6.5 Augmented activations as described in terms of the components of the augmented interactions framework in Chapter Three. 98

6.6 Augmented dragging as described in terms of the components of the augmented interactions framework in Chapter Three. 99

A.7 Augmented Interactions Questionnaire Results 151

LIST OF FIGURES

2.1 The IBM ScrollPoint Mouse includes an isometric joystick that is used for rate based scrolling. 13

2.2 The Logitech WingMan Force-Feedback Mouse is a mouse that provides vibrational haptic feedback to the user. The product's primary target was gamers. 14

2.3 (*Left*) The Inflatable Mouse facilitates limited haptic feedback and a continuous DoF input by squeezing the mouse [39]. (*Right*) The Rockin' Mouse provides an additional DoF by monitoring mouse tilt [5]. 16

2.4 (*Left*) The Nintendo Wii Balance Board is a new pressure-based input device. (*Right*) Several of the Playstation 2 controller's buttons are able to sense how hard a user is pressing. 17

2.5 The Nintendo DS includes a pressure-sensing screen and facilitates stylus based interactions. 19

3.1 Elements of the conceptual framework for augmented interactions. 29

4.1 (*Left*) A uni-pressure augmented mouse with a sensor in the Top location for the second finger. (*Right*) Our dual-pressure augmented mouse design with sensors located in the Top location for the second finger and in the Left location for the thumb. 41

4.2 (*Left*) The top sensor location for the second (middle) finger. (*Middle*) The left sensor location for the thumb. (*Right*) The right sensor location for the third (ring) or fourth (little) finger. 43

4.3 Visual approximations of three discretization functions that were tested in our research. DF1 is a linear distribution function. DF2 and DF3 are quadratic distributions centered at the highest pressure level and the middle pressure level respectively. 44

4.4 Visual examples of the first study's stimuli. (*Left*) The user moved the red pointer to the blue target. (*Right*) When the target was selected correctly, the target's color changed from blue to yellow. 49

4.5 Location and distribution of targets and their actual centers in terms of pressure units at varying numbers of pressure levels. D1, D2, D3 and D4 are the relative target distances used. W indicates the width in pressure units for each target. 50

4.6 Mean trial completion time, by sensor location and number of pressure levels. Note that the y-axis begins at 1s to better highlight differences. 53

4.7 Mean number of crossings per trial, by sensor location and number of pressure levels. 53

4.8 Mean trial completion time, by uni-pressure selection mechanism and number of pressure levels. Note that the y-axis begins at 0.5s to better highlight differences. 54

4.9 Mean number of crossing per trial, by uni-pressure selection mechanism and number of pressure levels. 54

4.10 An example categorization of targets into coarse-level and fine-level items for the *switch-to-refine* and *tap-and-refine* techniques. 58
4.11 Visual examples of the second study's stimuli. (*Left*) The user moved the red group highlight and the red pressure cursor to the blue target. (*Right*) When the target was selected correctly, the target's color changed from blue to yellow. 60
4.12 Mean trial completion time, by dual-pressure selection mechanism and number of pressure levels. Note that the y-axis starts at 0.5s to better highlight differences. 62
4.13 Mean number of crossings per trail, by dual-pressure selection mechanism and number of pressure levels. 63
4.14 Two traces of user pressure data when performing a uni-pressure selection task with the *click* selection mechanism. 64

5.1 (*Left*) The left mouse button is disabled by taping it down and a pressure sensor is affixed and used as a pressure button. (*Right*) A pressure sensor is installed underneath the left mouse button casing creating a pressure-augmented mouse button. 67
5.2 With *pressure tap* a mouse click is invoked by applying pressure beyond a minimum pressure threshold and releasing pressure within a time interval of T. In our implementation a T of 150 ms and a minimum pressure threshold of 2 units were used. 70
5.3 With *pressure click* a mouse down is invoked by applying pressure beyond a threshold of P_{down}. A mouse up is invoked by releasing pressure below a threshold of P_{up}. In our implementation a P_{down} of 4 pressure units and a P_{up} of 2 pressure units were used. 71
5.4 With *hard press* a double-click is activated by applying pressure beyond a threshold that is significantly higher than the *pressure click* single-click threshold. With *pressure click* two consecutive pressure clicks are required to enter a double-click. 73
5.5 Mean completion time, by selection technique and location. Standard error bars are also included. Note that the y-axis begins at 0.36 seconds to better highlight differences. (BC: Button Click, PC: Pressure Click, PC+A: Pressure Click Audio, PT: Pressure Tap, PT+A: Pressure Tap Audio.) 77
5.6 Number of errors, by selection technique and location. (BC: Button Click, PC: Pressure Click, PC+A: Pressure Click Audio, PT: Pressure Tap, PT+A: Pressure Tap Audio.) 78
5.7 Pressure by time plot of two typical single-click actions with *pressure tap* and *pressure click.* 79
5.8 Mean completion time, by action-invocation technique. Standard error bars are also included. (BC: Button Click, PC: Pressure Click, PT: Pressure Tap, HP: Hard Press, HP+A: Hard Press Audio.) 83
5.9 Number of errors, by action-invocation technique. (BC: Button Click, PC: Pressure Click, PT: Pressure Tap, HP: Hard Press, HP+A: Hard Press Audio.) 83

5.10	State transitions for common mouse operations with a pressure-button mouse described using Buxton's three state model [15].	89
6.1	a) An augmented scroll button in use. b) As pressure input increases, the document scroll rate increases.	92
6.2	a) An augmented scroll thumb in use. b) As pressure input increases, the document zooms out, and the scroll thumb expands accordingly.	94
6.3	a) An augmented preview of a file folder. b) As pressure input increases, the preview thumbnail size increases.	95
6.4	a) An augmented back button. b) As pressure input increases, additional pages are traversed. Visual feedback shows the user what pages the user can traverse back to.	97
6.5	a) A file folder that can be activated with an augmented activation. b) When pressure input crosses a threshold the folder is opened.	98
6.6	a) The user has configured a folder to only accept image files. b) When pressure input crosses a threshold, the file shrinks to indicate it can be moved, overriding the default.	100
A.1	Consent Form	127
A.2	Pressure Augmented Mouse Experiment 1 TLX and Questionnaire	129
A.3	Pressure Augmented Mouse Experiment 2 TLX and Questionnaire	135
A.4	Pressure Buttons Experiment 1 TLX and Questionnaire	139
A.5	Pressure Buttons Experiment 2 TLX and Questionnaire	142
A.6	Augmented Elements TLX and Questionnaire	145
A.7	Augmented Elements Questionnaire Results	151

List of Abbreviations

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
DoF	Degree-of-Freedom
GUI	Graphical User Interface
HCI	Human Computer Interaction
N	Newtons of Force
WIMP	Windows, Icons, Menus and Pointers

CHAPTER 1

INTRODUCTION

In the past 30 years desktop computer interactions have changed very little. The newest desktop interfaces like Microsoft Windows Vista[1] and Apple's OS X Leopard[2] look and act similarly to the Xerox Star [10], following the direct-manipulation Windows, Icons, Menus and Pointers (WIMP) paradigm [62] and making use of a keyboard and two-button mouse for input. In particular, interactions with the main elements of the user interface – that is, widget controls, object icons, and actions such as drag-and-drop – are still much the same as those designed for the first graphical interfaces [10]. While designs based on the WIMP model have obviously been successful, they also have a number of flaws [6, 7, 8, 10, 34, 36]. For instance, WIMP interfaces often require a large number of widgets, with each widget typically mapped to a single system command. As a result, higher-level tasks like navigating and searching are not well supported, requiring multiple controls to be activated, or a single control to be activated multiple times, in order to accomplish a real-world task. For example, the high-level task of navigating a document is poorly supported by WIMP interfaces [4, 31] because navigational subtasks like scrolling and zooming are controlled by separate widgets.

To improve support for higher-level tasks, standard desktop interactions must be augmented or redesigned with the high-level tasks of users in mind. Some improved WIMP interactions have been designed [4, 21, 31, 40, 42, 56, 57]; however, these more expressive interactions often require one or more additional degrees of freedom from input devices. Expressive interactions could be designed for desktop interfaces as well, but more powerful and expressive input must be made available before these improvements can be made.

Take for example the tablet PC: while desktop interactions have suffered from the lack of expressive input devices, tablet PC interactions have become more powerful and

[1] Windows Vista `http://www.microsoft.com/windows/windows-vista/`
[2] Apple OS X Leopard `http://www.apple.com/macosx/`

expressive, and new interactions that push the limits of traditional WIMP interfaces have been developed [1, 23, 41, 49, 55, 57, 60, 69]. These innovations can partly be attributed to the pressure-sensing capabilities of the tablet PC's stylus; whereas the mouse offers only two dimensions of continuous input, a pressure-sensing stylus allows both two-dimensional pointer control and the additional dimension of pressure input. Pressure input has also been employed successfully in other contexts [12, 17, 58, 68], and can increase user efficiency and expressive control. Considering this evidence, it is possible that desktop interactions could also be made more powerful and expressive if pressure input were available.

A number of avenues could be explored for adding continuous pressure input to desktop interactions, but augmenting the mouse for pressure sensitivity may be the most practical option; mice have been successfully augmented in a number of ways over the years [2, 5, 29, 39, 54, 62, 70] and some augmentations, such as the scroll-wheel, have been adopted by mainstream users [29]. However, while these mouse augmentations have been well tested, augmenting the mouse with pressure-sensitive input has not yet studied. Pressure-sensing interactions have been tested in the context of some devices [12, 17, 58, 68], but to our knowledge no research has been performed to inform the design of a pressure-augmented mouse. Even the most basic design issues such as ideal sensor placement, ideal number of sensors, and number of controllable pressure levels remain unexplored. This thesis carries out research to answer some of these basic design questions and to test pressure-augmented mouse interactions in a desktop context.

In summary, before desktop interfaces can be made more expressive through pressure-sensing interactions, the design space of the pressure-augmented mouse must be explored and tested, and empirical data to inform the design of pressure-augmented mice must be collected and analyzed.

1.1 Problem

The problem we address in this thesis is that there is a lack of ergonomics and performance information for the design of pressure-augmented mice.

While pressure input could be offered to the user through a number of devices, the mainstream acceptance of the mouse and the success of other mouse augmentations [29, 70] suggest that the mouse may be the best platform for offering pressure-based input to desktop users. However, a number of questions regarding ergonomics and human performance must be answered before a pressure-augmented mouse can be deployed. For instance, the most effective locations for applying pressure on a mouse are unknown, and the number of pressure levels controllable with a pressure-augmented mouse is unknown. These and other fundamental design questions must be answered.

There are two possible configurations for pressure-augmented mice that can be explored: One configuration calls for the addition of pressure-sensitive buttons to one or more locations on the mouse. Some research has shown that users can control multiple points of pressure with the same hand [54, 58] but it is uncertain whether users will be able to simultaneously control multiple sensors on a mouse. If users are able to manipulate more than one sensor, the number of controllable pressure levels could be increased. Ideally, this mouse configuration should allow users to control continuous pressure input without inhibiting other common tasks like mouse button clicking.

The second mouse configuration would see the standard mouse buttons themselves replaced or augmented with pressure sensitivity, adding expressive pressure input by providing more powerful standard buttons. If pressure input can adequately replace clicking and double-clicking functionality, similar to the way touchpads simulate clicking on laptops [3, 9, 44, 53, 61] then pressure-sensitive input could be added to the mouse without cluttering the device with additional buttons or sensors. In addition, users would not be required to move their fingers to acquire the pressure buttons.

1.2 Solution

Our solution is to provide empirical performance and ergonomics information in the design of a pressure-augmented mouse.

To provide this information we tested two pressure-augmented mouse configurations in quantitative experiments. The first configuration called for one or two sensors to be installed on a standard mouse. Design parameters for this configuration, including three possible sensor locations, were tested using a discrete target selection task similar to a task employed for testing pressure-sensitive pens [55]. Empirical results and subjective rankings were used to judge the most appropriate designs, and a set of design recommendations for future pressure-augmented mice were produced. The second configuration involved replacing standard mouse buttons with pressure sensors. Design parameters for this configuration, including several techniques designed to simulate single and double-clicks, were tested using timed single-click and double-click tasks. Empirical results and subjective rankings were used to judge the most appropriate designs, and a set of design recommendations for future pressure button mice were produced. Later, we also test one of our pressure-augmented mouse designs in the larger context of facilitating more expressive desktop interactions by developing six pressure-controlled desktop interactions and by performing a subjective user study.

1.3 Steps in the Solution

Develop a framework for interaction

Our research to design an effective pressure-augmented mouse is only relevant in the larger context of providing more expressive desktop interactions. Developing a framework to describe how interactions can be made more expressive is the first step in understanding how to improve desktop interactions and where a pressure-augmented mouse fits in the overall context of interaction. Our augmented interactions framework, presented in Chapter Three, was used to identify the scope of our main research and to design the six augmented desktop interactions that were tested in the last step.

Determine what pressure-augmented mouse configurations to test

We consulted human factors research and previous research with pressure-based interactions to identify a set of locations where pressure sensors could be installed on a mouse. These possible sensor locations and the high-level design goals for augmented mice led to two possible mouse configurations: the first, with one to two sensors installed on the mouse in addition to the standard buttons; and the second, with sensors replacing or augmenting the standard mouse buttons. The first configuration called for testing three sensor locations, while the second configuration called for testing two locations.

Answer basic performance and ergonomics questions

By consulting previous research with pressure-sensing interactions we chose a set of low-level test tasks that could be used to compare parameters of the pressure-augmented mouse configurations. In total, five experiments were carried out. The first configuration, with sensors installed on the mouse exterior, called for two experiments in order to test uni-pressure and dual-pressure input on a mouse. The second configuration, with sensors replacing buttons, called for three experiments: one to test single-click techniques, one to test double-click techniques and one to test double-click techniques with a pressure-augmented button variation of the second configuration. Using empirical evidence from these experiments, answers to the following questions were determined:

What design parameters are most effective for each configuration?

How many levels of pressure can users control?

Can dual-pressure input increase the number of controllable pressure levels?

Can users perform clicking operations using a sensor instead of a button?

Can pressure control be improved with a linearization function?

What level of feedback do users require to control pressure?

Test an augmented mouse design in real-world interaction

To evaluate the pressure-augmented mouse as a solution to the larger problem of expressivity in desktop interfaces we implemented a set of six pressure-controlled desktop interactions. We carried out a subjective user study of these augmented interactions to determine if the pressure-augmented mouse improved user power and expressivity in common desktop tasks.

1.4 Evaluation

To provide information about performance and ergonomics for pressure-augmented mice, we performed five experiments to test design parameters for two different pressure-augmented mouse configurations. By analyzing the results of each experiment we were able to identify the most effective design parameters and provide design recommendations for pressure-augmented mice. In addition, to show that a pressure-augmented mouse increases expressive control in realistic desktop tasks, we performed a subjective evaluation of six augmented interactions. Results of the evaluation show that users find the pressure-augmented interactions to be desirable, easy to learn, and easy to use.

Our evaluation took place in three main parts, each discussed in a chapter of this thesis:

1. Mice augmented with pressure sensors were evaluated using a discrete target selection task [55]. Both uni-pressure and dual-pressure mice were produced and tested. Empirical results from the experiments were used to determine the number of controllable pressure levels, an effective linearization function, the most effective sensor locations and the most effective selection mechanisms for discrete target selections.

2. Mouse buttons were replaced with pressure sensors and evaluated using single-click and double-click tasks. We compared the use of sensors in two locations and compared five single-click techniques and four double-click techniques. The most effective single and double-click simulation techniques and feedback requirements were determined using empirical evidence as well as subjective user feedback. A quantitative study comparing double-click mechanisms using a pressure-augmented button was also performed.

3. A subjective evaluation of six desktop-based augmented interactions was performed using a mouse with a pressure-augmented primary button. An exit questionnaire was used to gauge user reaction to the pressure-based interactions.

1.5 Contributions

There are two primary contributions presented in this thesis: the results of our experiments in Chapters Four and Five that provide performance information for the design of pressure-augmented mice, and the design recommendations for pressure-augmented mice and pressure buttons that were developed from the results of our experiments.

The secondary contributions of this thesis are the design framework for augmented interactions that can be used to develop interactions for the pressure-augmented mouse as well as for other devices and contexts (Chapter Three), the six augmented interactions that were developed and tested (Chapter Six), the identification of seven factors that can affect performance with a pressure-augmented mouse (see section 4.1), and the design of two novel dual-pressure interactions (see section 4.4.1).

1.6 Thesis Outline

Chapter Two presents a survey of related research and products which form the foundation for the research presented in this thesis. First, the human capacity to sense and apply pressure is discussed. Second, we discuss a number of mouse augmentations that have been performed for research and consumer products. Third, discrete and continuous pressure-based interactions for a variety of devices are discussed. Fourth, we discuss continuous and discrete interaction techniques that are similar to pressure-control techniques. Fifth, we describe and discuss pressure and touch-sensing selection techniques that have been developed for research and commercial products. Finally, WIMP interactions that have been augmented are discussed.

In Chapter Three we set up our research with pressure-augmented mice by introducing a design framework for developing more expressive, augmented interactions. Using this framework, we motivate our research of pressure-augmented mice as a solution to the problem of poor expressivity in desktop interactions. In addition, the framework we present can be used to compare and contrast interactions, and to design interactions for a number of input devices. In Chapter Six we use this framework to design six augmented interactions for the pressure-augmented mouse.

Chapter Four presents our work to investigate ergonomics and performance questions for pressure-augmented mice. The factors that can affect performance with a pressure-augmented mouse are identified and two user studies are carried out to answer important design questions. Both uni-pressure and dual-pressure augmented mice are studied, and the most effective design parameters for each are identified. Design recommendations for future pressure-augmented mice are presented.

Chapter Five presents our work to investigate ergonomics and performance questions for pressure-sensor buttons – that is, pressure sensors installed to replace standard mouse buttons. Four pressure-based button click techniques and four pressure-based double-click techniques are designed and tested in two user studies. Extensions of the pressure button designs are developed and discussed, and an informal study of pressure-augmented mouse buttons is performed. Design recommendations for future pressure buttons are presented.

Chapter Six presents an application of our work with pressure-augmented mice. To determine if pressure-augmented mice can improve desktop interactions, we designed and implemented six augmented desktop interactions: standard graphical user interface (GUI) controls and objects that have been augmented with additional or more expressive functionality and controlled with a pressure-augmented mouse. Six augmented interactions were developed using the design framework presented in Chapter Three. A subjective study of the interactions is performed, and results of the study are discussed.

Chapter Seven presents a discussion of the most important results from Chapters Four, Five and Six. Higher-level implications of our findings and issues related to the work as a whole are addressed. Lessons that have been learned over the course of our work are discussed.

Chapter Eight summarizes the research presented in this thesis, discussing the main contributions of our work and highlighting avenues of future work that have been opened as a result of this thesis.

CHAPTER 2

RELATED WORK

To augment the mouse with pressure-sensitive input and design augmented GUI interactions that take full advantage pressure-input capabilities, six areas of research literature must be examined.

First, since we are designing a new pressure-based input device for human users, we must understand the human capacity to apply and sense pressure; any design for a pressure-augmented mouse must conform to these specifications. Experiments performed by researchers in the fields of psychology and physiology provide detailed knowledge of the human systems involved along with their capabilities and limitations.

Second, we must examine commercial products and research projects that have augmented the mouse with secondary input devices. Since our goal is to augment the mouse rather than design a new input device, our design must conform to the general form factor of a mouse and provide pressure-sensing functionality. Understanding other augmented mouse designs will aid us in designing a mouse that unobtrusively supports pressure input, allowing both pressure-sensing interactions and standard mouse operations to be performed without conflict.

Third, we must examine other pressure-input tools and devices to identify the properties to evaluate in our pressure-augmented mouse and to survey the additional functionality that pressure input can provide. Pressure input studies will give insight into the types of interaction that are best performed with pressure and what test tasks should be performed with our pressure-augmented mouse.

Fourth, we must discuss non-pressure-based interactions that could be controlled through pressure input. Pressure-sensitive input, like other input mechanisms, has both strengths and limitations that make it suited to certain types of interactions. Examining interactions that are similar to pressure-based interactions will give insight into the best uses of pressure-based mouse input.

Fifth, in order to assess the viability of a mouse with pressure sensors replacing the mouse buttons, we examine other input devices and tools that use pressure to simulate button clicks. Using pressure input for button click tasks has been studied in a number of other research papers and their results will inform our design process.

Finally, to identify possible end uses for the pressure-sensing functionality of our augmented mouse, we will discuss research projects that have developed more expressive interactions for WIMP interfaces. Some research projects have demonstrated that more expressive input can be leveraged through augmented GUI widgets and controls. This interaction paradigm allows the new input functionality to be integrated through familiar operations of the system, while allowing previously learned interactions to remain.

2.1 Human Control of Pressure

Our design for a pressure-augmented mouse relies on the ability of human users to apply and control pressure with their fingers on a mouse form factor. Pressure sensors must be installed on the mouse such that users are able to apply and control pressure as easily as possible. Regardless of the form factor, all pressure-based computer input relies on two key human abilities: the ability to sense pressure, and the ability to apply pressure.

2.1.1 Sensing Pressure

Commonly referred to as the sense of touch, the ability to sense pressure is only one part of the larger somatosensory system: the network of sensors in human skin responsible for sensing pressure, temperature, body position (proprioception), and pain (nociception) [38]. The sense of pressure is referred to as the sense of tactition. This sense is made possible through different types of mechanoreceptors: sensory receptors that respond to touch, pressure, and movement on the skin [38, 50]. There are four well known types of mechanoreceptors, all four of which can be found in the palms and fingers of humans [50, 51]. Most important to pressure-sensing mouse interactions are the Meissner's corpuscles, which respond to light touch, and the Pacinian corpuscles, which respond to changes in pressure applied to the skin [38, 50]. Both of these corpuscles are described as "rapidly adapting," meaning that they send signals to the brain at the beginning and end of a sensation only [38, 51]. Other tactition sensors include Merkel's discs and Ruffini corpuscles, which respond to changes in skin tension resulting from textured objects and moving objects respectively, and are important for maintaining grip [50, 51].

All of these sensory receptors will respond when a user is interacting with a pressure-augmented mouse, but the Pacinian corpuscles are likely the most important. This is because sensing pressure and applying pressure form a closed feedback loop, and Pacinian corpuscles allow the user to sense the magnitude of the pressure they are applying. The sense of proprioception, the awareness of body position and muscle use, is also important for identifying the magnitude of pressure applied. Proprioceptive feedback provides awareness of the amount of force applied through the muscles [38]. These two senses grant the user the ability to sense the amount of pressure they apply to a pressure-augmented mouse.

2.1.2 Applying Pressure

The amount of pressure a person can apply depends on several factors. These factors include the muscle groups involved, the point of the body from which pressure is applied, the position of the body, and the age and sex of the person [32]. While various points of the body can apply pressure, a pressure-augmented mouse requires applying pressure using the hand. Previous research has shown that the fingertips are the best location for applying pressure using the hand [67]. Therefore, pressure-based interactions with a computer mouse should employ the tips of the fingers and/or the thumb while the hand is spread over the form factor of a mouse with the arm in a relaxed position.

A study conducted by Imrhan and Loo [32] tested the limits of pressure application for the thumb and fingers with the hand and arm positioned similarly to when using a mouse. Participants were asked to apply pressure on a Preston pinch meter in five configurations: a pinch-like action between the thumb and each of the four fingers, and a fist-like squeeze between the thumb and side of the first finger. Their results show that for both males and females the ordering of strength is thumb, first finger, second finger, third finger, fourth finger, in decreasing order of strength [32]. The highest average force, in Newtons (N), applied by adult males was with the thumb at 92.18 N (standard error 3.4 N) and the lowest average force applied by adult males was with the fourth finger at 28.44 N (standard error 1.9 N). The highest average force applied by adult females was with the thumb at 63.74 N (standard error 1.7 N) and the lowest average force applied by adult females was with the fourth finger at 19.61 N (standard error 1.0 N) [32].

2.2 Mouse Augmentation

For a pressure-augmented mouse design to be successful, the sensors must be installed in locations on the mouse that are accessible and usable. Additionally, the augmentation must not interfere with standard mouse operations. Such augmentations have been performed successfully in the past, including the addition of buttons to the mouse's form factor, the addition of tactile or haptic feedback, and the inclusion of extra degree-of-freedom (DoF) devices.

2.2.1 Additional Buttons

Manufacturers continue to add additional buttons to the mouse's form factor. Some designs have included multiple secondary buttons on the left and right sides of the mouse as well as on the top of the mouse. Adding additional buttons can make certain tasks easier but doing so requires a user to remember the mappings between buttons and functions, and may require that fingers be repositioned to facilitate input. Buttons on the sides of the mouse may be accidentally depressed during normal mouse movement and clutching (picking up and repositioning the mouse when moving the cursor long distances).

One particularly successful augmentation is the scroll wheel, usually installed on the top of the mouse and accessible by the first and second fingers. The scroll wheel is a variation of a button that allows for discrete input along a single bidirectional axis. Most commonly used as a surrogate for scrolling tasks, the scroll wheel allows users to scroll vertically or horizontally in a window without moving the mouse cursor to activate the scroll bar. Some studies have shown the scroll wheel to be particularly effective when navigating through long documents [29, 72].

Similar in function to the scroll wheel, the IBM ScrollPoint[3] is a mouse that features an isometric joystick rather than a scroll wheel. Like the scroll wheel, the joystick is installed on the top of the mouse and is accessible by the first and second fingers. Pressure applied to the joystick controls the rate and direction of scrolling in a window. Like the scroll wheel, the joystick on the IBM ScrollPoint gives the user an additional bidirectional DoF.

Figure 2.1: The IBM ScrollPoint Mouse[3] includes an isometric joystick that is used for rate-based scrolling.

The TrackMouse [47] is 2D + 2D controller, allowing two axes of control like a standard mouse and an additional two axes of control from a small trackball mounted on the top of the mouse in place of a scroll wheel. Martin and Raisamo [47] performed experiments comparing the TrackMouse to bimanual control of two mice in a two-cursor control task. Their results show that users were somewhat slower using the TrackMouse than when using two mice [47], however the TrackMouse gives the user four degrees of freedom with a single-handed interaction.

[3] IBM ScrollPoint `http://www.almaden.ibm.com/u/zhai/topics/scrollpoint.htm`

2.2.2 Force Feedback

The tactile mouse [2] is a standard computer mouse that has been augmented with a small actuator, allowing the mouse to vibrate under certain conditions. This form of feedback can inform the user when certain events are occurring, such as when the cursor is moving into different areas of a window or when the user is crossing window boundaries. Akamatstu and colleagues [2] conducted a study to compare the effects of tactile feedback, visual feedback and auditory feedback in mouse-based selection. Their results show that users complete selection tasks better with tactile feedback over visual and auditory conditions [2].

Similar to the tactile mouse, the Logitech WingMan Force-Feedback Mouse[4], a now discontinued commercial product, allows for directional force feedback along the mouse's two axes of motion. To our knowledge this type of directional feedback has not been tested in experiments, but could allow for richer, more meaningful haptic feedback in situations similar to those tested by Akamatstu and colleagues [2].

Figure 2.2: The Logitech WingMan Force-Feedback Mouse[4] is a mouse that provides vibrational haptic feedback to the user. The product's primary target was gamers.

[4] Logitech WingMan Force-Feedback Mouse `http://www.amazon.com/Logitech-WingMan-Force-Feedback-Mouse/dp/B00001W01Z`

2.2.3 Additional Degrees of Freedom

The Inflatable Mouse [39] is a volume-adjustable mouse, compact enough to fit inside the PC card slot of a laptop. Before the mouse is used a balloon on the inside of the mouse is inflated, allowing the form factor to be adjusted to the size of a standard mouse. The Inflatable Mouse also offers some more powerful input and output options than a standard mouse. The balloon inside the mouse is fitted with a gas-pressure sensor, allowing the user to squeeze or apply pressure to the mouse and control continuous parameters. The mouse can also provide some limited haptic feedback to the user in that the balloon can be expanded and deformed. The mouse is also equipped with two touch sensors used as the primary and secondary mouse buttons, and an array of touch sensors that take the place of a scroll wheel [39].

The Rockin'Mouse [5] is a mouse with a unique form factor that has been augmented with tilt-sensing accelerometers. This mouse has a rounded bottom, allowing users to tilt the mouse along the control surface and use the tilt sensors as an additional DoF. Balakrishnan and colleagues [5] used the additional control provided by the tilt sensors in three-dimensional (3D) object positioning tasks, allowing the mouse to be used as a 2D and 3D pointer. Results of their experiments show that users were up to 30% faster when using the Rockin'Mouse for 3D object positioning tasks [5]. Also providing an extra DoF, MacKenzie and colleagues [45] designed a mouse with two tracking balls on the underside. This modification allows software to capture the angular movement of the mouse along the z-axis. The angular motion of the mouse is calculated using the relative displacement of the data from the two mouse balls, facilitating rotation tasks without mode switching.

The VideoMouse [27] is a mouse with a video camera installed on the underside. The VideoMouse software runs a real-time vision algorithm that calculates 6DoF mouse movement by comparing camera images over time. The mouse is able to sense two axes of horizontal motion like a standard mouse, tilts of the mouse forward, backward, left and right, rotation of the mouse around the z-axis and limited height sensing. As a result, the VideoMouse facilitates a number of 3D manipulation tasks. Using an augmentation similar to the VideoMouse, Siio and colleagues [65] introduced the FieldMouse which augments the mouse with a visual ID recognizer. Using the FieldMouse users can interact with virtual objects using any flat surface that has an embedded ID strip, such as a barcode in a book.

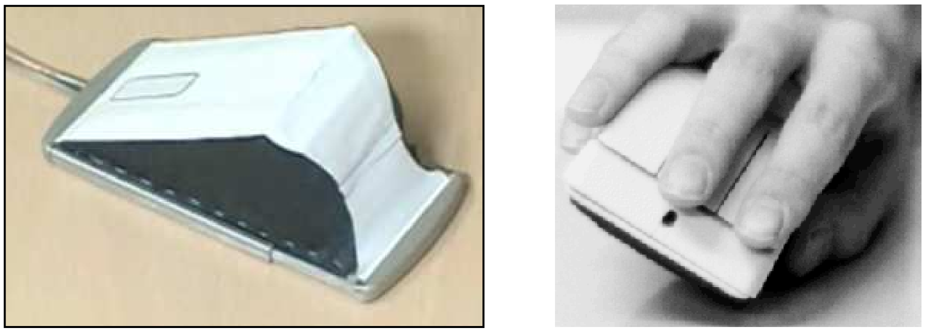


Figure 2.3: (*Left*) The Inflatable Mouse facilitates limited haptic feedback and a continuous DoF input by squeezing the mouse [39]. (*Right*) The Rockin' Mouse provides an additional DoF by monitoring mouse tilt [5].

2.3 Pressure Interaction

Pressure-based interactions have become more common in research and commercial products due in part to the greater availability of pressure-sensing devices. These include pressure-sensing pens like those found in tablet PCs, pressure-sensitive buttons like those found in the PlayStation 2 controller[5], and other pressure-sensing devices like the Nintendo Wii Balance Board[6]. Numerous studies have proposed novel interaction techniques or investigated different applications and offer guidelines for working with pressure-based input [12, 17, 55, 57, 58, 68].

Like all input devices, pressure sensors can be discussed in terms of their sensing properties. Common input device properties have been identified in previous research on input devices [16, 28, 35]. As discussed in these articles, a pressure sensor is an absolute, continuous, reflexive, single-DoF input device. As such, pressure sensors can be used for both discrete and continuous interactions. We discuss these input device properties in greater depth in Chapter Three.

[5] Playstation 2 `http://www.playstation.com`
[6] Nintendo Wii Balance Board `http://wii.com`

Figure 2.4: (*Left*) The Nintendo Wii Balance Board[6] is a new pressure-based input device. (*Right*) Several of the Playstation 2[5] controller's buttons are able to sense how hard a user is pressing.

2.3.1 Discrete Pressure Interaction

Discrete pressure interactions involve dividing the continuous pressure input space into a number of discrete levels. Discrete selections using pressure have been studied in the form of target selection tasks, though no studies with pressure-sensing mice have been performed and little research has been done with controlling more than one pressure sensor. Several researchers have developed discrete interaction techniques using pressure and have explored different uses for discrete pressure interaction.

Ramos and colleagues [55] explored the design space for discrete pressure-based interactions with a pressure-sensing stylus and identified that users' capacity to control discrete pressure values is tightly coupled to the number of discrete pressure levels, the type of selection mechanism, and the degree of visual feedback. Their results show that users can effectively control a maximum of six discrete levels of pressure when given full visual feedback about their current pressure level, and a maximum of four discrete pressure levels when given no visual feedback. Their results also show that users are fastest with a QuickRelease selection mode, where the user quickly lifts the stylus from the tablet PC to indicate selection. Their results are mainly applicable to single-DoF pressure input with a stylus.

Other research investigating user pressure input capabilities includes the work of Mizobuchi and colleagues [49] who conducted a study investigating user accuracy and control when performing discrete, pressure-based selection tasks using a stylus. Their results show that continuous visual feedback is better than discrete visual feedback, that users can best control forces that are less than 3.0 N, and that 5 to 7 discrete pressure levels are appropriate for accurate discrimination and control of input values [49]. Like the work done by Ramos and colleagues [55] their results are mainly applicable to stylus-based pressure input, and their work does not explore pressure input with multiple sensors.

Pressure input can be used in parallel with other forms of input, allowing simultaneous control over two aspects of an interaction. Ramos and Balakrishnan [57] developed a technique for a pressure-sensing stylus called Pressure Marks, where pressure input is used to specify an action and stylus movement is used to select an object or group of objects to perform the action on. Their technique used only two discrete levels of pressure but monitored changes in pressure over the course of a selection, allowing a total of four different actions to be specified using their technique. Experimental results show that specifying targets and actions in parallel significantly reduces task completion times [18, 57].

Some research has been done to test more limited forms of pressure input. Zeleznik and colleagues [70] proposed an additional *pop-through* state to the mechanical operation of mouse buttons. These pop-through mouse buttons have two depressed states, soft-press and hard-press, allowing a number of interaction techniques to take advantage of multiple active states. Using pop-through buttons, Forlines and colleagues [21] proposed an intermediary *Glimpse* state to various buttons in application GUIs. With Glimpse, users can preview the effects of an editing action before the action is performed. Multilevel interactions similar to Glimpse could be used to improve user control in navigation, editing and selection tasks.

Other work using only two discrete pressure levels includes work with pressure-sensitive touchpads like those found in laptop computers and portable devices like the Nintendo

DS[7]. Blasko and Feiner [12] proposed dividing touchpads into multiple pressure-sensitive strips, each used primarily as a position-based slider. Pressure sensitivity allows for two “layers” of sliders, one accessed with light touch and the other accessed by applying additional pressure to the touchpad. Results of their experiments show that two-level pressure interactions do not require visual feedback [12], which is consistent with the work done by Ramos and colleagues [55].

One of the challenges associated with pressure-sensing interactions is the lack of bidirectional input. Rekimoto and Schwesig [58] propose a solution to this using a touchpad-based pressure-sensing device called PreSenseII that recognizes finger position by measuring the contact area and pressure applied. PreSenseII achieves bidirectional control of pressure input by identifying the position of the finger when pressure is being applied [58]. Two distinct finger poses are mapped to the directionality of the pressure input, allowing for both ‘negative’ and ‘positive’ pressure to be applied.

Figure 2.5: The Nintendo DS[7] includes a pressure-sensing screen and facilitates stylus-based interactions.

2.3.2 Continuous Pressure Interaction

Continuous pressure interactions involve mapping the continuous input space of the pressure sensor to some variable or set of variables within a system. As a continuous

[7] Nintendo DS `http://www.nintendo.com/ds`

input modality, pressure is well suited to controlling continuous variables [28, 35]. Several commercial products make use of such interactions and several researchers have developed other interaction techniques using continuous pressure input.

Isometric joysticks, like those found in the IBM ScrollPoint and some laptop keyboards, use pressure input to control the speed of the mouse cursor. Users decrease or increase the amount of force on the pointing stick to control the velocity of the mouse cursor. Similarly, the PalmMouse[8] is a handheld pointing device that allows users to control the cursor by manipulating a navigation dome on the top of the PalmMouse with their thumb. By varying the amount of pressure applied to the navigation dome the user can control the speed at which the cursor moves.

The continuous input capabilities of isometric joysticks have also been used for gestural input techniques. Wobbrock and colleagues [68] developed an alternative text-entry technique for mobile phones using an isometric joystick. The Edgewrite [69] technique, a gestural text-entry technique designed for stylus input, was modified for use with an isometric joystick. Though this new technique proved successful for text-entry on mobile phones, the authors offer no general analysis of human capabilities with pressure-based input.

Similar to Pressure Marks [57], pressure input can be used in a continuous form to control a parameter while performing another action. *Zliding* [56] is a technique that uses continuous pressure and the position of a stylus to simultaneously zoom and scroll. With the Zliding technique, pressure input zooms a document or image while the cursor position is used to scroll [56]. Ramos and Balakrishnan [56] indicate that the same technique could control any parameter with pressure input while moving the stylus. Although their technique was only tested using a pressure-sensing stylus, Zliding could be modified to work with a number of pressure-sensing devices.

Continuous pressure input can also be used on a large scale for tracking real world interactions. Srinivasan and colleagues [66] developed a modular system of pressure-sensitive floor mats to assist in dance instruction, physical rehabilitation, and digital motion capture applications. Their system delivers a high resolution of continuous

[8] PalmMouse `http://www.infogrip.com`

pressure input over a wide area with each individual sensor occupying an area equal to 6mm x 6mm, and running at a refresh rate of 30 Hz [66].

2.4 Non-Pressure-Based Interaction Techniques

A number of interactions have been designed that add expressivity through an additional DoF. Many of these interactions are analogous to pressure-based techniques and some could theoretically be controlled using pressure input. Although we do not discuss these interaction techniques in depth, they provide a context for our work to improve desktop expressivity through pressure-augmented mouse interactions. Here we discuss some interaction techniques that are similar to pressure-based interactions, and compare and contrast the input modalities used with the capabilities of pressure-based input.

Eye Gaze

Eye-S [52] is a system of input using gaze tracking hardware. The system tracks relative eye movements and absolute eye position, allowing the eyes to control pointer movement, to input commands, and to write using a Graffiti-like [71] gesture set. Tracking eye movement in this manner allows the eyes to be used for 2DoF bidirectional input. Two pressure sensors could be used to provide similar control, but bidirectionality can be difficult to achieve without mode switching or the use of an extra input variable [58]. This lack of bidirectionality also contributes to difficulties using pressure for relative movements, making pressure unsuitable for performing Eye-S system gestures [52]. Similar in design to the Eye-S system, Lucas and colleagues [43] developed interaction techniques that employed user gaze as an extra DoF to assist in resizing 3D objects and found that users were significantly faster resizing objects when using the combination of gaze and pointer control.

Voice

The human voice can be employed as a multi-DoF continuous input device. Pitch is a DoF of the human voice that is similar to pressure, in that the pitch of a sound produced by the vocal cords is modified by pressure from muscles that control the vocal cords. Harada and colleagues used voice input to control parameters such as line width in a 2D drawing program [26], a parameter that can already be controlled by stylus pressure in most drawing programs. *Voicepen* [26] is a system of input that allows the voice to control the movements of an on-screen cursor. Each vowel sound is mapped to a direction on the screen, giving the user relative control over cursor position [26]. Pressure input could be used in a similar way by mapping pressure input of one sensor to the direction of cursor movement, and input from another sensor to cursor velocity.

Bimanual Input

Benko and colleagues [9] developed several techniques for controlling continuous parameters like control-display ratio through bimanual interactions on a multitouch screen. Though their techniques are designed as bimanual interactions, similar continuous parameter control can be facilitated through pressure input without requiring bimanual input which can sometimes be cumbersome. Kabbash and colleagues [37] studied the impact that bimanual interaction has on compound task performance and found that bimanual interactions can have both positive and negative effects on performance. Their results indicate that certain kinds of bimanual interactions, where the second hand's action is dependent on the first, can yield the highest performance when the interaction technique is designed properly [37].

Modes

Some interactions make use of additional parameters without requiring an additional input device. For instance, modes are a common way to increase interactive power without adding extra degrees of input freedom. The FlowMenu [25] is a type of marking menu [73] that makes use of multiple modes, allowing the user to select commands, set parameters, and perform text entry with a stylus. However, modes can increase complexity and confusion by using a single device for multiple separable interactions.

Time

Another way to increase the power of an interaction technique without requiring extra hardware is to use time. Time is used in acceleration functions for rate-based controls, to control activation through dwell time [13, 55], and as a dimension in gestural input techniques like Pressure Marks [57]. However, using time as a dimension of the interaction implicitly removes some user control over the interaction, and users may take longer performing a task if they are waiting for dwell timers to expire and acceleration functions to reach their peak velocity.

2.5 Alternative Selection Interactions

Here we discuss a number of interactions that have been used instead of button clicking to indicate selection. Some research suggests that pressure input could be used for clicking and double-clicking [44, 53]. However, replacing or augmenting standard mouse buttons with pressure sensors may have unintended negative effects on usability and performance throughout the system; some evidence suggests that selection actions like button clicking can consume a significant amount of the total time taken during pointing tasks [13, 30]. These alternative selection interactions have been divided into two groups: pressure-based selection, and non-pressure-based selection.

2.5.1 Pressure-Based Selection

Pressure input has been explored by several researchers as an alternative to button clicking, which we refer to as *pressure clicking*. Pressure clicking has been proposed for the mouse [70], for touchpads [44, 61], for mobile phone text-entry [17], and for touch screens [9].

Using a Synaptics touchpad, MacKenzie and Oniszczak [44] facilitate pressure clicking by giving users aural and haptic feedback when the touchpad is pressed and released. To prevent spurious clicks, transitions from clicking to releasing include hysteresis, meaning that the pressure level that maps to a button down action is greater than the pressure level that maps to a button up action. However, MacKenzie and Oniszczak [44] do not provide the pressure levels they used to simulate the button clicks and instead suggest that thresholds must be determined empirically.

Pressure clicking has been employed as an alternative to the standard multi-tap technique for cell phone text-entry [17]. In such systems only a limited number of pressure levels (three or four) are necessary to enter text with each key [17]. Clarkson and colleagues also discuss the possibility of using discrete and continuous pressure input to perform zooming or scrolling tasks in large workspaces [17].

A pseudo-pressure clicking technique, SimPress, was implemented for facilitating precise selection techniques for a multitouch screen [9]. Benko and colleagues [9] map changes in the finger's contact area to changes in pressure. SimPress requires users to perform a small rocking movement with their finger, rolling their fingertip backwards towards their wrist to simulate a click. Benko and colleagues were able to achieve fairly accurate selection rates on a touchscreen using this mechanism [9].

2.5.2 Non-Pressure-Based Selection

Designers have proposed several alternatives to button clicking for performing selection actions. Some of these interactions have been found to be faster than button clicking. For instance, Bohan and Chaparro [13] compared a mouse click to a dwell-to-click interaction. In their study Bohan and Chaparro found that a hover of 200 ms provided an improvement of as much as 25% for task completion times in comparison to a mouse button press and release [13]. A similar selection interaction is currently available with the GentleMouse[9], a commercial product designed to eliminate button clicks. With the GentleMouse users hold the mouse cursor still for a configurable duration to initiate a click. After the duration has elapsed, a small trigger window appears. A button click is simulated by moving the mouse cursor into the trigger window and holding the cursor still again. The GentleMouse may be helpful for users with repetitive strain injuries; mouse clicking has been found in some cases to aggravate disorders such as carpal tunnel syndrome [19].

When using a stylus, users commonly invoke a selection action by directly tapping and then releasing the stylus over an object. Because tapping does not reflect how people naturally use pens and notepads, where writing and making checkmarks is common, Ren and Moriya [59] developed an alternative referred to as *touching*. Unlike tapping, which requires the pen to briefly touch the screen and then be lifted off to select an item, touch

[9] Gentle Mouse `http://www.gentlemouse.com`

interactions only require that the target be touched at some point in pen movement. This touch-based selection technique supports selecting targets by crossing them, by making checkmarks, and by tapping. Results show that touching is a viable alternative to tapping [33, 59], even for the elderly [33]. These results are consistent with a similar study which compared a target-crossing selection technique called CrossY with standard point-and-click selections [24]. CrossY was found to be more efficient than point-and-click selections because the technique combines pointing and selection in one fluid motion.

Touchscreens facilitate one of the most natural forms of pointing and selecting by allowing users to touch and select virtual objects with a finger. Potter and colleagues [53] compared three selection mechanisms: take-off, first-contact and land-on. The take-off technique allows the user to drag a cursor that appears above the user's finger tip and select an object by removing their finger from the touchsceen when the cursor is over the intended target. The first-contact technique allows users to drag their finger across an empty area of the touchscreen and select an object by making contact with it, similar to the stylus-based *touching* technique used by Ren and Moriya [59]. The land-on technique triggers selection the first time the finger lands on the screen. Their results show that users perform best with the take-off selection technique [53]. This is in contrast to the results presented by Ren and Moriya [59], though they did not test a selection technique like take-off in their study. Albinsson and Zhai [3] extended the work of Potter and colleagues [53] to design more accurate selection mechanisms on touchscreens. However, their research primarily focused on reducing pointing errors on touchsceens instead of comparing selection mechanisms.

2.6 Augmented GUI Interactions

Once the pressure-augmented mouse has been designed and tested, the problem of finding a suitable application for pressure-sensitive mouse input remains. While an absolute, continuous, reflexive input device like a pressure sensor could be used for a number of interaction techniques, designing augmented GUI interactions which respond to pressure input is an attractive option for a number of reasons. Augmented GUI interactions allow pressure input to be used in a number of contexts for a variety of purposes, maximizing the benefit of our augmentations. Using augmented GUI interactions also integrates pressure input with the GUI system allowing for visual learning of the pressure interactions. Implementing augmented GUI interactions also adds new functionality to the standard GUI system by making common GUI elements more powerful while allowing familiar functionality to remain. Most importantly, augmented GUI interactions can solve the overall problem of poor expressivity in current WIMP-based GUIs. A number of researchers have presented augmented GUI interactions that make the interface more expressive. Here we discuss previously designed augmented interactions that make use of pressure as well as other forms of input.

Using pressure input from a stylus, Ramos and colleagues [55] designed and tested GUI widgets controlled by discrete pressure levels. Though they suggest that similar widgets could be developed to respond to continuous pressure, their analysis was limited to widgets controlled by discrete stylus pressure. In their discussion they include several general widget designs for both continuous and discrete pressure input modalities. The pressure widgets they present could be implemented for a number of pressure-sensing devices, including a pressure-augmented mouse.

Some augmented GUI interactions have been designed to make use of multilevel discrete input devices, like pop-through buttons. The Glimpse [21] system, mentioned earlier, provides users with a lightweight method for exploring possible system states. Though their implementation used a multi-state pop-through button, Glimpse states are similar to two-level pressure widgets [55] and could be controlled by any discrete device with sufficient levels. Results of experiments with pressure-sensing devices also suggest that two-level widgets could be controlled by a pressure-sensing device with no additional visual feedback [55].

In addition to pressure input, tablet PC pens also allow for a “tracking” state where the pen is moved above the screen but not touching it. *Hover widgets* [23] are augmented GUI widgets designed to respond to gestures from a stylus in the hover state over a display. Users can click and activate widgets normally, and by moving the stylus in the tracking state other actions can be activated by performing gestures with the stylus. As their techniques were gesture-based, they are only used for discrete selections. Using a stylus, both Pressure Widgets [55] and Hover Widgets [23] could theoretically be implemented together.

Rather than using input from a DoF, more complex object behaviours can be implemented by using system constraints. Agarawala and Balakrishnan [1] explored the design space of a virtual desktop where objects were subject to constraints through a physics simulation. Objects could be given size and mass and required more effort to move due to friction and gravity. This allowed for some more complex object interactions, such as tossing, piling and sorting that were intended to leverage the capabilities of a tablet PC stylus [1].

CHAPTER 3

MOTIVATION: DESIGNING MORE EXPRESSIVE INTERACTIONS

In this chapter we introduce a framework for describing and designing more expressive interactions to highlight the importance of designing more expressive input hardware like pressure-augmented mice. While the majority of this thesis is focused on the design of pressure-augmented mice, such research is only relevant in the context of developing improved, more expressive augmented interactions. This framework can also be used as a tool for designers of pressure-augmented mouse interactions; in Chapter Six, we test a number of augmented desktop interactions that were designed using the framework described in this chapter.

3.1 A Framework for Augmented Interaction

Here we present a conceptual framework for augmented interactions that is based on a high-level view of a user's interaction with a GUI (Figure 3.1). The framework has at its core the idea of an interaction, which we define as a combination of an object in the interface with one or more actions, each of which have a characteristic degree of freedom. Interactions are undertaken in service of a user task, and are supported by input mechanisms that provide the actual input data. In the following sections we describe each part of the framework in more detail, starting with the idea of an interaction.

3.1.1 Interaction: Object + Actions

In a WIMP interface, an interaction can be defined as a user's manipulation of an on-screen entity. We formalize this with the concepts of the GUI object and the interface action; using these concepts, an interaction can be specified as one or more actions applied to an object.

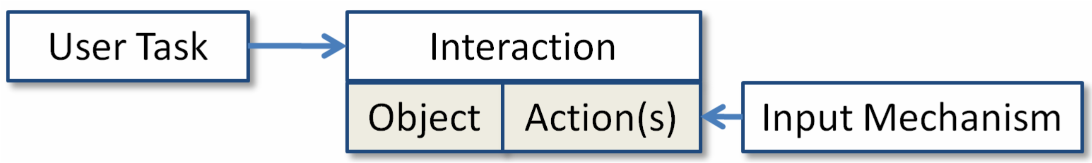


Figure 3.1: Elements of the conceptual framework for augmented interactions.

WIMP Objects: Data and Controls

An object is an entity in a WIMP interface that has a visible representation. There are two basic object types in WIMP-based GUIs: data objects and controls.

- *Data objects* are the visual representations of the data of interest – icons in a file explorer, text and links in a web browser, and custom objects in an application (e.g., entities in a UML diagram editor). Data objects are the 'nouns' of direct manipulation [63].

- *Controls* are graphical instruments that allow manipulation of data [6]. Since controls lie between the user's actions and the actual data, they are indirect instruments in a GUI. Traditional widgets such as buttons and sliders are the most common examples of controls; however, some types of data objects can also be given control capabilities (such as the links on a web page, which act both as data on the page, and as buttons that invoke navigation actions).

Actions in WIMP interfaces

Actions are the manipulations that are possible with a data object or control. Actions can be characterized by the degrees of freedom of the data that is being manipulated.

- *1D-Discrete*: The action is used to specify one of multiple states along one dimension. For example, selecting an icon in a file browser implies being able to specify which of two states the icon is in. 1D-D actions are often implemented with two-state devices such as mouse buttons, but devices with more than two states can also be employed [70].

- *1D-Continuous*: These actions allow specification of a single continuous value. For example, scrolling a document with a scroll thumb is a 1D-continuous action. 1D-C actions can receive input from devices that are one-dimensional, but can also use a single dimension of a 2D device (e.g., 1D scrolling using a 2D mouse).

- *2D-Continuous*: These actions allow simultaneous specification of two dimensional continuous values. An example action is 2D movement of a cursor; input is commonly received from any of several 2D pointing devices.

- *Higher-dimensional actions*: 3D and higher-degree actions are needed in some applications, but are not common in WIMP interfaces. We do not consider these actions further, other to note that high-degree-of-freedom input devices may have extra dimensions that could be used in the augmentations described below.

Higher-level manipulations can be specified using these action primitives. For example, the common GUI idiom of dragging can be characterized as an interaction made up of two actions: a 1D-D action (to select the object) plus a 2D-C action (to move it across the screen). Similarly, the idiom of 'Shift-clicking' can be characterized as a combination of two 1D-D actions: one for the shift, and one for the click.

Note that while a 1D-D action can occupy only a limited number of states, 1D-C actions could be viewed as 1D-D actions with a large number of discrete states. To our knowledge there is no definitive standard for the number of states that an interaction requires in order to be considered continuous; however, a continuous interaction must be perceived as continuous by users, and not as collection of discrete states.

3.1.2 Augmentation

An augmentation is a modification that is made to an action to increase expressive power. Based on the primitives defined above, there are several possible augmentations.

- *Adding states to a 1D-Discrete action*: A simple augmentation involves increasing the number of states that are possible for an interaction: for example, adding a state to an on-screen button changes it from a two-state widget to a three-state widget. Pop-through buttons [70] and the Glimpse technique [21] are examples that have been explored in previous research.

- *Adding a 1D-Discrete action to an existing action*: Adding a discrete dimension to an existing action allows a multiplication of the expressiveness of the original – essentially adding modes to the interaction. Examples include Shift-clicking and Control-dragging, which are now common in commercial GUIs, or research techniques such as Pressure Marks [57], which uses discrete pressure levels during a drag operation to change the behaviour of the operation.

- *'Upgrading' a 1D-Discrete action to 1D-Continuous*: Allows the conversion of state-based manipulations to continuous manipulation. For example, a scroll button uses a 1D-D action; changing to a 1D-C action allows the scroll button to support variable-rate scrolling [6] (assuming an appropriate 1D-C input source).

- *Adding a 1D-Continuous action to a 1D-Discrete action*: This augmentation can allow a continuous-value specification at the same time as a discrete selection. For example, Benko and colleagues developed techniques for continuous parameter control using finger position on a multitouch screen with bimanual interactions [9].

- *Adding a secondary 1D-Continuous action*: Multiple dimensions can be controlled simultaneously with the addition of other 1D-C actions. For example, OrthoZoom [4] adds support for zooming (a secondary 1D-C action) to an existing 1D-C action (scrolling). Note that adding a second 1D-C action need not convert the interaction to a true 2D manipulation (e.g., horizontal and vertical scrolling); rather, it can remain a composite of two 1D manipulations [34] (as with OrthoZoom [4]).

- *Adding a 1D-Continuous action to a 2D-Continuous action*: There are many ways that 2D movement can be augmented with an additional degree of freedom. For example, 1D-C pressure sensitivity is already used to control line thickness in tablet PC drawing applications; pressure has also been used to control cursor size [60] and to zoom during 2D pointer movement [56].

- *Adding a 2D-Continuous action to a 2D-Continuous action*: These augmentations add a second 2D capability to an interaction. Current examples generally involve the addition of a second 2D position controller – as seen in multitouch displays which allow multiple fingers to simultaneously move, rotate, and scale objects.

At a minimum, an input mechanism must meet the dimension requirements of the interaction. However, it should be noted that higher-dimension input can be used for lower-dimension actions. For example, a 1D-C input could be quantized to provide a 1D-D action. This happens frequently when time is used as an input dimension; several techniques define a time threshold that allows for two or more discrete states to be specified. For example, ‘hover help’ combines the 2D-C action of pointing with the 1D-D action of ‘pausing for a certain period,’ which is a quantized version of the time dimension.

As stated earlier, an interaction is made up of a GUI object and a set of actions. By adding to or modifying the actions related to an object, extra dimensions are added to the interaction which must be controlled by some input mechanism. In the following section we discuss input mechanisms as they relate to actions, and later discuss some additional rules for pairing input mechanisms and actions.

3.1.3 Input Mechanisms

Although a variety of input sensors and mechanisms can be used to control augmented actions, not every device is suited to every action, and choosing an input device for an action is more complex than simply pairing input devices and actions by the dimensions they control. The following paragraphs set out some of the important issues in matching an input device to an action.

Input Mechanism Properties

Pressure input, like all other input modalities, can be described in terms of input device properties. The properties of the input mechanism can guide the pairing of input mechanism and action, and here we highlight five properties that have been identified in previous research on input issues (e.g., [16, 28, 35]).

- *Physical Property Sensed*: Common properties sensed by input devices include position and force. Positional devices generally map best to positional tasks, and force has traditionally been used as a mapping for rate [28, 35]. However, exceptions to this rule can be found: the mouse is used for rate control in Microsoft Windows, and pressure-sensing devices have been used for single-DoF positional control [55].

- *Absolute vs. Relative Mapping*: Absolute devices like sliders and pressure sensors have a fixed 'zero' location (as with pressure sensors), whereas a mouse and scroll wheel only sense relative movements. Relative devices are advantageous because they can be mapped to very large virtual spaces; however they also require 'clutching' (having to reposition the device when moving long distances). Absolute devices are best mapped to finite virtual spaces [28, 35].

- *Continuous vs. Discrete Input*: Continuous devices like mice, foot pedals and pressure sensors map best to continuous tasks, but can also be mapped to discrete selections depending on the desired granularity [28, 35]. Discrete devices usually provide the user with physical affordances, such as mechanical 'clicks' and detents.

- *Reflexivity*: This is a property of absolute devices like pressure sensors and isometric joysticks; these devices return to their 'zero' position when released by the user. Reflexive devices avoid the ‘nulling’ problem [14] that can occur when an input device is set for one action and then another action is begun with the device not reset to the default ‘zero’ location.

- *Bidirectionality*: This is a property of relative devices like mice and scroll wheels; input can be specified as both positive and negative along a single axis. However, bidirectionality can be implemented with absolute devices by including a mode switch [58], a second sensor, or by defining the mid-point of the sensor to be the ‘zero’ location (e.g., joysticks).

As defined by our framework, pressure sensors are single DoF, absolute, continuous, force-sensing input devices with the reflexivity property.

Sources of Input for Increasing Expressivity

This thesis is primarily concerned with augmenting interactions using pressure input from a pressure-augmented mouse; however, the input for an augmented interaction could come from other devices as well. In situations where additional devices are impractical to add to the system, other input schemes can be employed. We have identified five ways that additional input capability can be obtained:

- *Overload the existing input capability with modes*: In this scheme, adding a 1D-D DoF facilitates a mode switch for another input. For example, holding down a modifier key (such as Shift or Control) could change the behaviour of continuous actions (e.g., scrolling pages instead of lines with the scroll wheel) or discrete actions (e.g., open a link in a new tab instead of in the current window). FlowMenu [25] for example makes use of modes to increase the input capabilities of a stylus.

- *Leverage time as a DoF*: In this scheme, time is used as a 1D-D or 1D-C DoF. To get discrete input, time can be quantized (e.g., ‘hover help’ activates after a time delay); time can also be used as a continuous parameter for acceleration functions (e.g., scrolling accelerates the longer a scroll button is activated), or for mode switching (e.g., the difference between two clicks and a double-click). Time is commonly used in WIMP interfaces, and many gestural input systems use time as a DoF.

- *Use constraints*: In this scheme constraints are added to an interaction in order to create more complex behaviour. For example, Kruger and colleagues [40] developed a constraint-based system that allowed objects to be rotated and positioned using a standard 2D mouse. In another example Agarawala and Balakrishnan [1] gave data objects size and mass in a virtual desktop system with gravity and friction, allowing for more complex object behaviours and interactions. In another example, Shoemaker and Gutwin [64] developed techniques for automatically setting zoom level based on the idea of maintaining the visibility of several control points.

- *Leverage unused degrees of freedom*: In this scheme an unused DoF in the input device is used to control the augmented action. For instance, Zliding [56] leverages the unused pressure DoF to control zooming while sliding or scrolling with a stylus.

- *Add new degrees of freedom*: The approach advocated in this thesis is to add new input capabilities to the input device to provide the needed degrees of freedom. Some upgrades take an existing device and transform it into a higher-DoF device, as with the 6DoF VideoMouse [27]. Other upgrades to devices come in the form of independent input devices, as with the addition of the scroll wheel. Degrees of freedom can also be added to a system through other modalities, including user gaze [43], bimanual input [37] or continuous voice input [26]. In particular, we are interested in exploring the addition of pressure input to the mouse.

3.1.4 User Task

Although specific tasks for augmented interactions will vary, there are several general reasons for wanting additional expressiveness during an interaction. We have identified four in particular:

- *Integrate interactions that make sense together or are part of a higher level task*: In some situations, additional tasks can be naturally combined with existing tasks. For example, scrolling and zooming [4, 34], or rotation and translation [42] are naturally combined into single navigation actions.

- *'Working with your hands full:'* In some cases it is important to provide alternate mechanisms for interaction when a primary mechanism is in use. For example, 'spring-loaded' folders[10] in the Mac OS allow users to open folders while dragging a file (using time as an added 1D-D control, since the mouse button is in use).

- *Integrate multiple single actions into a continuous control*: Frequent and repetitive single actions can often be reconsidered as continuous manipulations; for example, multiple presses of a 'Back' button could be converted into a multilevel 'Reach Back' button that goes back a variable distance. Ramos and colleagues' Pressure Widgets provide a similar interaction [55].

- *Allow richer input*: There are several situations where additional expressiveness could allow users to be more judicious in the execution of their tasks. Different types of richness include being able to express variable levels of selection (e.g., 'lightly selected,' 'strongly selected'), express variable levels of confidence in an action [21], or choose variable levels of preview. Many real-world examples exist – such as the way that the volume of a spoken command reflects its urgency: "open the door" versus "OPEN THE DOOR."

[10] 'Spring-loaded' folders `http://kb.iu.edu/data/aehp.html`

3.2 Applying the Framework

Although any number of augmented interactions are possible, not all augmentations would be effective or useful. When designing an augmented interaction, one can begin by describing the existing interaction in terms of the framework components: object, action(s), user task and input mechanism. By analyzing the interaction in terms of its parts, possible augmentations may reveal themselves. Comparing two similar augmented interactions in this manner can also reveal strengths and weaknesses in their respective designs, and potentially identify the more promising design. We have identified several issues that designers should consider when assessing the potential value of an augmented interaction:

- *Integrality vs. Separability*: When augmenting an interaction, it may be unclear whether a higher-DoF device is more suitable than two lower-DoF devices. The principles of Integrality and Separability [34] can assist when making this decision. Tightly coupled, integrated object properties (e.g., size and position) are best controlled with a single high-DoF input device, while separable properties (e.g., size and hue) are best controlled with two separate lower-DoF devices [34].

- *Leverage natural mappings*: How a device is used can sometimes map naturally to the interaction itself. For instance, the rotation dimension of a 6DoF tracker maps to the rotation of an object [42], stylus hover maps to 'above' the surface layers [23], and multi-state buttons can be used to indicate definiteness and confidence [21]. In addition, the direction of movement of the device and on-screen feedback should be compatible if possible [6].

- *Higher DoF is not always better*: Higher-DoF actions can be useful in some situations, but troublesome in others. For instance, 2D drawing is accomplished with a mouse or stylus, but drawing a straight line (1D drawing) is difficult. As a result, programs include modes for locking an input dimension (e.g., holding Shift allows straight lines to be drawn, a 1D-D augmentation). Even if the extra dimensions are not used, a device that matches the degrees of freedom of the action is better suited to the task [6, 34] (e.g., a 2D mouse performs better than a 6DoF device or two 1DoF devices in 2D pointing tasks).

- *Combine closely related interactions*: Some object parameters are naturally related (e.g., size and position, rotation and position) and suited to being combined in a single interaction [34]. OrthoZoom [4] combines scrolling and zooming, two aspects of document navigation.

- *Feedback*: All interactions should provide some form of feedback related to the state of the input control. Some absolute devices, like foot pedals and hardware sliders, already give some feedback to the user visually and through the user's proprioceptive sense; however, visual feedback presented on or near the augmented GUI object is also important since the user's visual attention is on the object at the time of activation. Visual feedback is particularly important for pressure-sensing devices [49, 55]. Feedback through other modalities such as haptics [48] and pseudo-haptics [46] has proven useful in some cases for promoting user awareness of GUI objects.

3.3 Relationship to Other Models and Paradigms

A number of models exist for designing and developing interactions, including Direct Manipulation [63], Instrumental Interaction [6], Reality-Based Interaction [36] and Post-WIMP interaction [7]. In addition, paradigms for interaction design have been identified [8]. Our augmented interactions framework is not meant to replace other design models; rather it is a tool for comparing and designing interactions that are developed in the context of other interaction models. Although we have presented this framework in the context of WIMP interfaces, our framework can be applied to other interface models.

For example, an interface like CPN/Tools [7] does not include scroll bars, pull-down menus, or the notion of selection. Instead the interface includes a number of Post-WIMP interactors like toolglasses, marking menus [73] and floating palettes as well as elements of direct manipulation. However, the augmented interactions framework could still be employed within this context: toolglasses could include multiple modes or their size could be modifiable with an augmented interaction, floating palettes include buttons that could be augmented, and the direct manipulation actions can also be augmented. Our framework could also be utilized in emerging interaction contexts based on Direct Manipulation [63] like Reality-Based Interaction [36].

3.4 Summary

Using this framework, we can identify opportunities where a pressure-augmented mouse could be used to make desktop interactions more expressive. In terms of sensing capabilities, a standard mouse provides a 2D-C relative pointing device and a number of 1D-D mouse buttons. A pressure-augmented mouse adds one to two 1D-C absolute, force-sensing devices. According to our framework, a number of augmented interactions are possible with this device combination: the 1D-C pressure input can be used to 'upgrade' existing 1D-D interactions, to control 1D-C parameters of existing 1D-D, 1D-C, and 2D-C interactions, and pressure input can be discretized and used as a multilevel discrete selector. Pressure input allows richer 1D input than mouse buttons, and also provides natural mappings to interactions like rate control [28, 35].

Based on this analysis, it is likely that a pressure-augmented mouse could be used to make desktop interactions more powerful and expressive. We now begin our primary research to provide basic ergonomics and performance information for pressure-augmented mice. Later, in Chapter Six, we use the principles of our augmented interactions framework to develop six augmented desktop interactions which we test in a subjective user evaluation.

CHAPTER 4

DESIGN GUIDELINES FOR A PRESSURE-AUGMENTED MOUSE

In this chapter we begin our main research to provide ergonomics and performance information for pressure-augmented mice. We identify the relevant design parameters for a pressure-augmented mouse and discuss research studies that we performed to answer key questions of performance and ergonomics. While pressure input has been studied with a number of devices, to our knowledge no research has been conducted with pressure-augmented mice; even the most basic questions, such as where to install pressure sensors on the mouse, have not been answered. A mouse with poorly placed sensors may restrict users to a limited number of pressure levels [49, 55], or make the mouse's pressure-sensing capabilities unusable. Poorly placed sensors may also interfere with standard mouse operations, like button clicking.

To effectively harness the potential of a pressure-augmented mouse, designers need to know the best locations for installing pressure sensors on a mouse, the number of pressure levels controllable with a pressure-augmented mouse, and what benefits, if any, multi-sensor augmentation has. Understanding the strengths and limitations of pressure input with a mouse can allow designers to augment the mouse with pressure sensors (Figure 4.1) and use the pressure-augmented mouse to facilitate more powerful and expressive desktop interactions.

In this chapter we set out to identify the most effective design parameters for pressure-augmented mice by performing two studies. In the first study we investigate the ideal locations for affixing pressure sensors to a mouse, the methods for selecting discrete pressure values, and the number of pressure levels that can be controlled with one sensor. To test the various design parameters we used a discrete target selection task, similar to a task that has been used to test pressure control with pressure-sensing pens [55]. The results of our first study show that users can best control pressure with their second (middle) finger and thumb. Our results also agree with previously established norms indicating that users can comfortably control only up to six pressure levels [49, 55].

In the second study we investigate user ability to control a larger number of pressure levels by utilizing two pressure sensors, and test two dual-pressure control mechanisms: *switch-to-refine* and *tap-and-refine*. Both of these dual-pressure control mechanisms facilitate control of 64 discrete levels. Results show that *tap-and-refine* is the most effective mechanism for selecting from a large number of discrete levels.

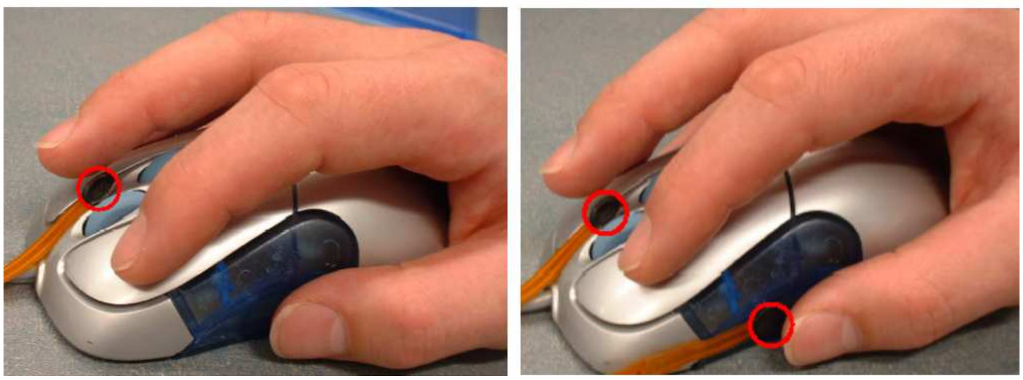


Figure 4.1: (*Left*) A uni-pressure augmented mouse with a sensor in the Top location for the second finger. (*Right*) Our dual-pressure augmented mouse design with sensors located in the Top location for the second finger and in the Left location for the thumb.

The main contributions presented in this chapter are:

1. Identification of the most effective design parameters for a pressure-augmented mouse, including sensor locations, controllable number of pressure levels, linearization function, and visual feedback.

2. Identification of factors that can influence performance with a pressure-augmented mouse in a discrete target selection task.

3. The design of two bidirectional, dual-pressure control mechanisms for pressure-augmented mice.

4. A set of design recommendations for future pressure-augmented mice.

4.1 Design of a Uni-Pressure Mouse

Here we identify the design factors that can influence performance with a pressure-augmented mouse in a discrete selection task. The seven factors that can influence performance are: number of levels, sensor position, number of sensors, discretization of raw pressure values, pressure control mechanism, selection technique and visual feedback.

4.1.1 Number of Levels

Although a pressure sensor is a continuous input device, the input can be discretized into a number of controllable pressure levels. Previous research with pressure input indicates that the number of pressure levels can have a significant effect on user performance [49, 55]. This research indicates that the maximum number of controllable pressure levels is six when users are given full visual feedback about their current pressure level [55]. To test for the number of controllable pressure levels with a uni-pressure mouse we tested 4, 6, 8, 10, and 12 pressure levels in our first study. In our second study, more targets were made available by utilizing pressure levels from two sensors, and we tested 4, 12, 16, and 64 targets.

4.1.2 Sensor Location

Designers can add pressure sensors to a mouse in multiple different locations. Ideally, pressure input should not require the user to interrupt a task or to reposition the hand to access a pressure sensor. Additionally, pressure control is best at the fingertips [66]. Therefore to provide greater user control and better resolution of pressure levels, designers should position the sensors so that they can be accessed within the reach of the fingertips.

The primary button on a mouse is typically controlled by the first (index) finger unless the button mappings are modified. Because the overarching design goal of this pressure-augmented mouse configuration is to add expressivity without altering standard mouse interactions, designers should not place a pressure sensor in a location that interferes with the first finger. Accordingly, for easy access and reduced task interruption, users should be provided access to pressure sensors through the thumb, second (middle) finger, and the third (ring) finger or fourth (little) finger.

Based on this analysis, three sensor locations were tested: top, left and right.

- *Top*: In the top location the sensor is positioned so that it can be easily acquired by the user’s second (middle) finger.
- *Left*: In the left location the sensor is positioned so that it can be acquired by the user’s thumb.
- *Right*: In the right location the sensor is positioned so that it can be acquired with the user’s third (ring) finger or fourth (little) finger.

Figure 4.2: (*Left*) The top sensor location for the second (middle) finger. (*Middle*) The left sensor location for the thumb. (*Right*) The right sensor location for the third (ring) or fourth (little) finger.

4.1.3 Number of Sensors

Most studies have investigated the use of pressure-based input on devices such as digitizers, pens or touchpads [49, 55, 58]. These devices are limited to a single-DoF input through pressure. However, we can affix one (uni-pressure) or two (dual-pressure) sensors onto the form factor of mouse so that users could acquire them simultaneously. In our first study we test the uni-pressure mouse, and in our second study we test the dual-pressure mouse.

4.1.4 Discretization of Raw Pressure Values

Analog-to-digital (AtoD) converters allow the force exerted on a pressure sensor to be interpreted as a raw stream of discrete numeric integer values. The analog force exerted by the user is converted to a digital data stream, usually a number from 0 to 256, 512, or 1024. However, users cannot control this many discrete values. As a result, applications further discretize the raw integer values by grouping nearby values into unique controllable pressure levels [49, 55].

Stylus-based pressure input studies have shown that users can comfortably control approximately six discrete pressure levels [49, 55]. Furthermore, users can best control forces that are less than 3.0 N [49]. However, there is no standard mechanism for discretizing the raw pressure values from a pressure sensor, and several methods and mappings of pressure levels have been used [49, 55]. Mizobuchi and colleagues [49] used a linear discretization function by creating equal pressure levels consisting of 0.41 N each. Ramos and colleagues [55] use a linear discretization function to map 1024 pressure values into equally distributed units.

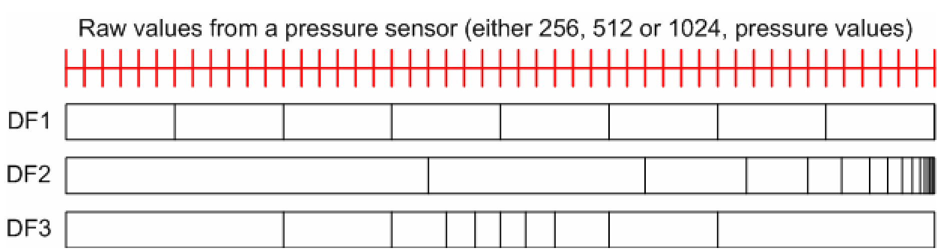


Figure 4.3: Visual approximations of three discretization functions that were tested in our research. DF1 is a linear distribution function. DF2 and DF3 are quadratic distributions centered at the highest pressure level and the middle pressure level respectively.

The discretization function used in our experiments was chosen by performing a pilot study with three subjects to compare three different pressure discretization functions: a linear function, a quadratic function centered at the highest pressure value and a quadratic function centered at the middle pressure value (DF1, DF2, DF3 in Figure 3.2). With the linear function we observed that users were less accurate selecting from low pressure values than from higher values. We found that users were fastest and most accurate with the quadratic function centered at the highest pressure value (DF2). In this discretization method, targets in the lower range contained more raw pressure units than those in the higher range.

4.1.5 Pressure Control Mechanism

A pressure control mechanism allows the user to iterate through a list of available pressure levels. With a uni-pressure augmented mouse, the pressure control mechanism is basic and simply consists of pressing down on one sensor to iterate through a limited number of pressure levels. However, with a dual-pressure augmented mouse, a pressure control mechanism must be designed in order to utilize both sensors. We propose that pressure control mechanisms with a dual-pressure augmented mouse consider the following design goals:

- The user should be able to access a larger number of pressure values than with one pressure sensor.
- There should be minimal overhead when the user switches between the sensors.
- Each pressure sensor should only be responsible for controlling a comfortable number of pressure levels.
- If possible, dual-pressure augmented mouse interactions should support bidirectional pressure control.

In section 4.4.1 we discuss two dual-pressure selection mechanisms that we designed: *tap-and-refine* and *switch-to-refine*.

4.1.6 Selection mechanism

A selection mechanism allows users to choose a desired value after using the pressure control mechanism to target a pressure level. Ramos and colleagues [55] proposed four selection mechanisms for stylus-based pressure input: QuickRelease, Dwell, Stroke and Click. With the QuickRelease mechanism, selection is indicated by quickly lifting the stylus from the tablet's surface after reaching the appropriate pressure level. With the

Dwell mechanism, selection is triggered after the user maintains pressure for a prescribed amount of time. With the Stroke mechanism, selection is indicated after the user makes a quick spatial movement with the stylus. With the Click mechanism, selection is indicated by pressing the stylus' barrel button. On a stylus, QuickRelease was shown to be the most effective selection technique [55]. However, it is not clear whether this method is appropriate for either a uni-pressure or dual-pressure mouse.

Based on our analysis, we chose three selection mechanisms to test with the uni-pressure augmented mouse: *quick release*, *dwell* and *click*.

- *Quick Release*: This technique is similar to the QuickRelease mechanism designed in [55]. In *quick release*, once the user reaches the desired target they select it by quickly releasing pressure from the pressure sensor.

- *Dwell*: This technique is similar to the Dwell mechanism designed in [55]. In this method the user maintains the cursor within the target for a predetermined amount of time. In our uni-pressure study we used a delay period of 1 second to trigger the selection, as done in previous research [55].

- *Click*: In this method the user maintains pressure over the desired target and clicks the left mouse button to select the item.

4.1.7 Visual Feedback

Kinesthetic feedback alone is insufficient for adequately controlling and selecting a pressure value [49, 55]. Visual feedback is a necessary component of the interaction space with pressure-based input [49, 55, 56]. The most common form of feedback is through a visual highlight over the active item that is selectable. Ramos and colleagues [55] investigated the effects of two different visual feedback conditions: full visual and partial visual feedback. In the full visual feedback condition the user's continuous level of pressure is visible, and a visual indicator (typically a highlight) iterates through the list of selectable items as well. In the partial feedback (or discrete feedback) condition only the visual highlight is shown. In a similar setup, Mizobuchi and colleagues [49] investigated the effect of continuous and discrete visual feedback. In both of these studies users performed better with continuous feedback.

4.2 Uni-Pressure User Study

To determine the most effective design parameters for our first pressure-augmented mouse configuration, we carried out two studies; one study of a uni-pressure augmented mouse, and one study of a dual-pressure augmented mouse. This first study was designed to identify the ideal sensor locations, number of pressure levels, and discrete selection mechanisms.

4.2.1 Hardware Configuration

Both studies used an optical mouse with pressure sensors mounted to the mouse casing. The sensors (model #IESF-R-5L from CUI Inc.[11]) could measure a maximum pressure value of 1.5 N. Pressure values were detected using a Phidgets interface kit [22] which converted the analog signal into 1000 raw pressure values. The experiments were conducted in full-screen mode at 1024×768 pixels on a P4 3.2 GHz Windows XP OS machine, and test applications were developed in C#.

4.2.2 Performance Measures

Experimental software recorded trial completion time, errors, and number of crossings as dependent variables. Trial completion time is defined as the time taken for the user to apply the appropriate amount of pressure and select the target. The number of crossings is defined as the number of times the cursor enters or leaves a target for a particular trial. The software records an error when the participant selects a location which is not a target. The trial ended only when the user selected the correct target, so multiple errors were possible for each trial.

[11] CUI Inc. `http://www.cui.com/`

The number of errors gives us a measure of success for the discrete selection task; however, the number of crossings provides a better indication of the level of control that users have, and is therefore a better overall metric for pressure control. This position is also supported by similar research [55]. As such, more time is spent in our discussion on the results of trial completion time and number of crossings. Participants were also asked in an exit questionnaire to rank the different selection mechanisms and sensor locations.

4.2.3 Methods

The main goal of this experiment was to determine the most effective location for installing a pressure sensor on a mouse and to determine the number of pressure levels controllable with a pressure-augmented mouse. The experimental design was adapted from a study used to test a pressure-sensitive stylus [55].

Participants

Nine participants (five males and four females) between the ages of 19 and 25 were recruited from the University of Saskatchewan for this experiment. All participants were students, all had previous experience with graphical interfaces and all used the mouse in their right hand.

Task and Stimuli

We used a serial target acquisition and selection task similar to the task in [55]. Participants controlled the movement of a red pointer through a sequential list of targets using pressure input. 900 of the 1000 raw pressure values from the sensor were discretized in a quadratic manner (see section 4.1.4). The targets were numbered and presented together as a vertical stack. During each trial one of the targets was coloured in blue. The user’s task was to apply sufficient pressure to move the red pointer over the blue target and then perform a selection action (Figure 4.3 *left*). We provided complete visual feedback to the user by highlighting each item as the user iterated through it. The color of the target changed blue to yellow and audio feedback was given when the task was completed correctly (Figure 4.3 *right*).

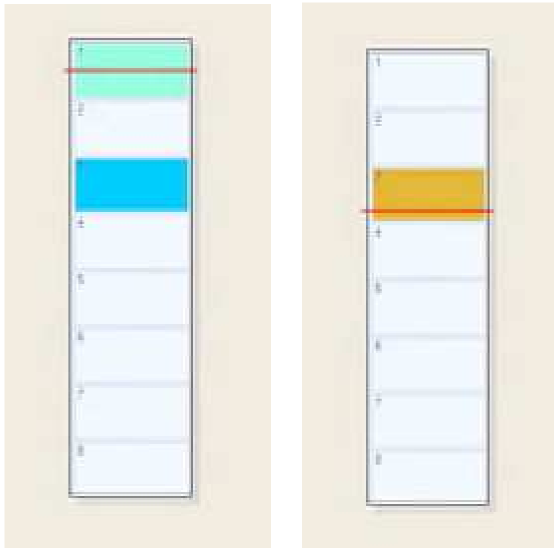


Figure 4.4: Visual examples of the first study's stimuli. (*Left*) The user moved the red pointer to the blue target. (*Right*) When the target was selected correctly, the target's color changed from blue to yellow.

Procedure and Design

The study used a 5×3×3×4 within-participants factorial design. The factors were:

- Pressure Levels: 4, 6, 8, 10, 12 (section 4.1.1).
- Sensor Location: Right, Left, Top (section 4.1.2).
- Selection Mechanism: Quick Release, Dwell, Click (section 4.1.6).
- Target Distance: 395, 535, 675, 815.

The order of presentation first controlled for *sensor location* and then for *selection mechanism*. Levels of the other two factors were presented randomly. At the beginning of the experiment we explained the selection mechanisms and participants were given ample time to practice. The experiment consisted of three blocks with each block comprising two repetitions for each condition. Targets tested were placed at four relative pressure distances. With nine participants, five pressure levels, three selection mechanisms, three sensor locations, four distances, three blocks, and two trials, the system recorded a total of (9×5×3×3×4×3×2) 9720 trials. The experiment took approximately 60 minutes per participant.

Target Distance

In each trial a target appeared at one of four different relative pressure distances: 395, 535, 675, and 815 pressure units (through the Phidgets interface kit [22], the sensor reported a value from 0 to 1000). These distances were chosen so that targets would occupy four unique pressure levels, no matter the total number of pressure levels in the trial. (Figure 4.4).

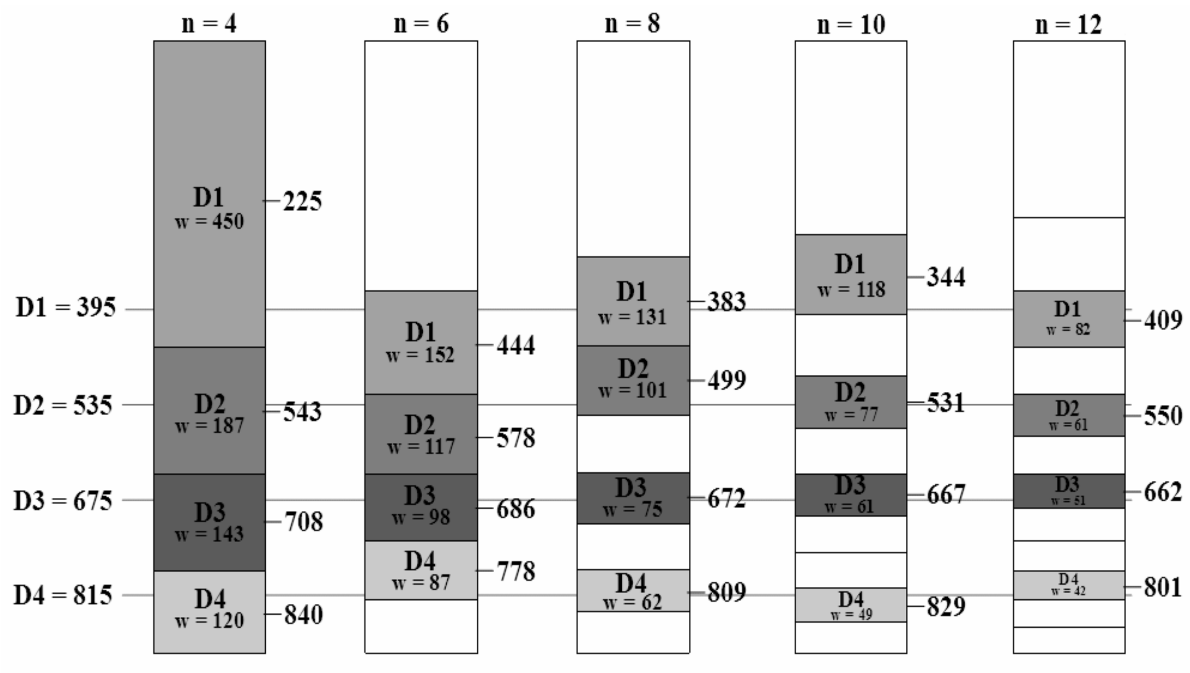


Figure 4.5: Location and distribution of targets and their actual centers in terms of pressure units at varying numbers of pressure levels. D1, D2, D3 and D4 are the relative target distances used. W indicates the width in pressure units for each target.

4.3 Results

Here we discuss the results of our first study with the uni-pressure augmented mouse, organized by dependent variable (trial completion time, crossings and errors, and subjective ranking).

4.3.1 Trial Completion Time

The overall mean completion time across all conditions was 1.97s (standard error = 0.023s). We used the univariate ANOVA test of trial completion time and Tamhane post-hoc pair-wise tests (unequal variances) for all our analyses. To make the data conform to the homogeneity requirements for ANOVA we used a natural-log transform on the trial completion time. Results showed main effects (all $p<0.01$) of trial completion time on *selection technique* ($F_{2,16}=20.05$), *sensor location* ($F_{2,16}=4.57$), *pressure levels* ($F_{4,32}=113.06$) and *target distances* ($F_{3,24}=21.655$).

Post-hoc pair-wise comparisons (using Tamhane post-hoc pair-wise tests, as noted above) of *pressure levels* yielded significant differences (all $p<0.01$) in trial completion time for all pairs except between pressure levels 4 and 6. Users were fastest when the number of levels was 4 and slowest when the number of levels was 12.
Post-hoc pair-wise comparisons of *selection techniques* yielded significant differences (all $p<0.01$) in trial completion time for all pairs. Participants were fastest with the *click* mechanism, followed by *dwell* and *quick release*. Figure 4.7 shows the mean completion time of each technique per pressure level.

Post-hoc pair-wise comparisons of *sensor location* yielded significant differences ($p<0.01$) in trial completion time between Right and Top, and Right and Left sensor pairs. Participants were faster with the sensor in the top sensor location followed by the left location and then right the location. Figure 4.5 shows the mean completion time for each sensor location across the different pressure levels.

Post-hoc pair-wise comparison of *target distance* yielded significant differences (all $p<0.01$) in trial completion time for all pairs except targets at relative distance D1 and D2.

4.3.2 Crossings and Errors

As identified by Ramos and colleagues [55], crossings are the most important metric for accuracy with pressure. While errors provide a measure of success for target selection tasks, crossings provide a measure for the overall controllability of pressure at various pressure levels. Therefore the majority of our analysis is performed on crossings rather than errors.

The average number of crossings per trial across all conditions was 1.3 (standard error = 0.022). An ANOVA test revealed a significant effect of *selection technique* on number of crossings ($F_{2,16}$=11.35, p<0.001).

Post-hoc pair-wise comparisons of *selection technique* yielded significant differences (all p<0.001) in number of crossings for all pairs. Click had the least number of crossings, followed by Dwell and Quick Release. Our tests did not show a significant effect of *sensor location* on crossings (all p>0.05). Figure 4.8 shows the average crossings per trial for each technique and Figure 4.6 shows the average crossings per trial for each sensor location.

The average number of errors across all conditions was 0.23 errors per trial (standard error = 0.007). With respect to *selection technique*, Dwell had the least number of errors (0.01) followed by Click (0.26) and Quick Release (0.42). For *sensor location* the ordering was Top (0.22), Left (0.23) and Right (0.25). The ordering of errors for *pressure level* was 4 (0.09), 6 (0.14), 8 (0.24), 10 (0.28) and 12 (0.41). The ordering of errors for *target distance* was D2 (0.22), D1 (0.24), D3 (0.23) and D4 (0.23).

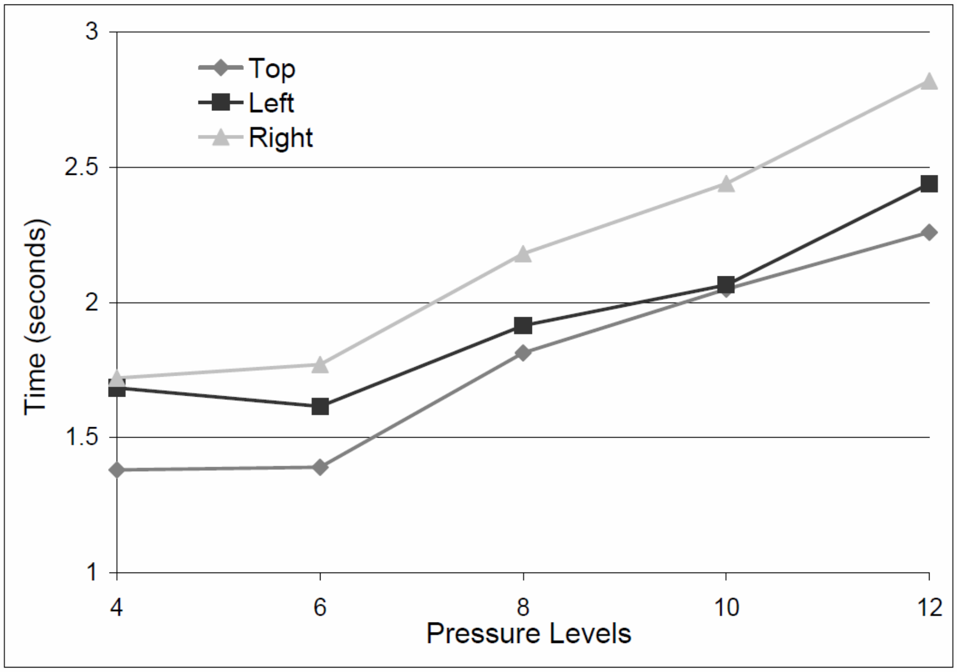


Figure 4.6: Mean trial completion time, by sensor location and number of pressure levels. Note that the y-axis begins at 1s to better highlight differences.

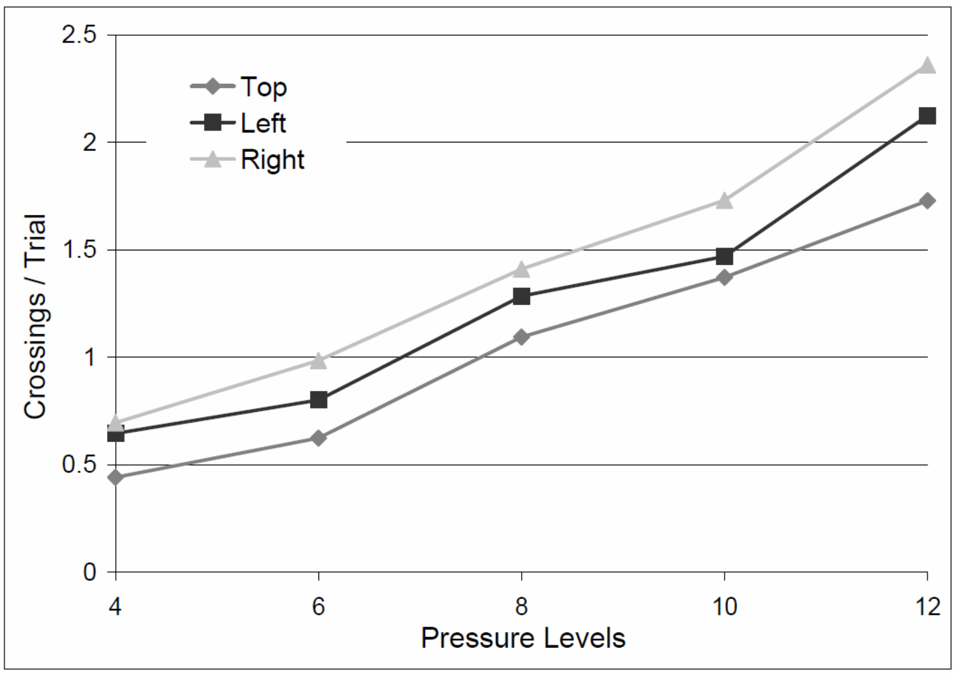


Figure 4.7: Mean number of crossings per trial, by sensor location and number of pressure levels.

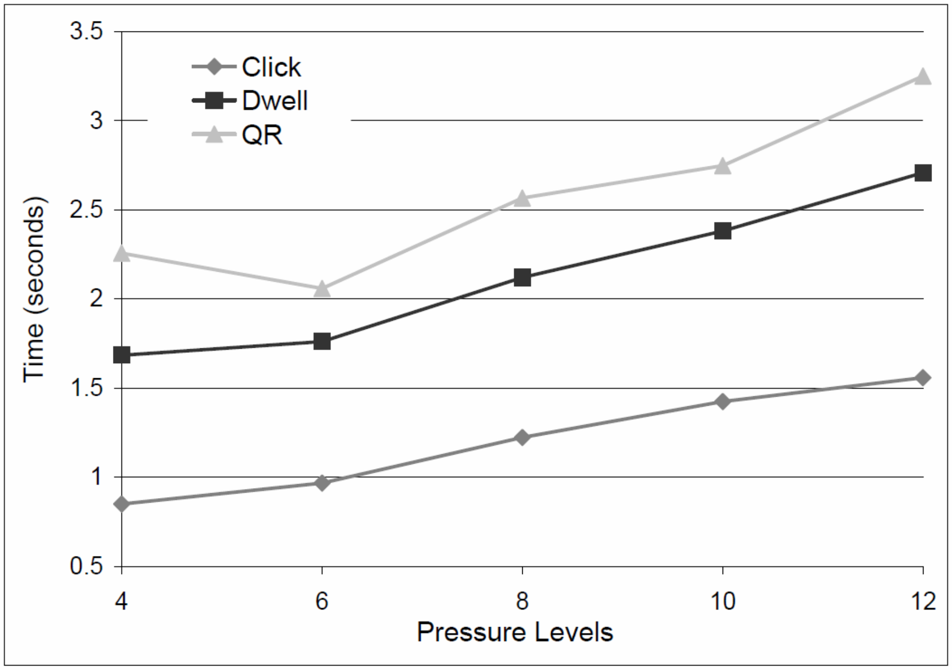


Figure 4.8: Mean trial completion time, by uni-pressure selection mechanism and number of pressure levels. Note that the y-axis begins at 0.5s to better highlight differences.

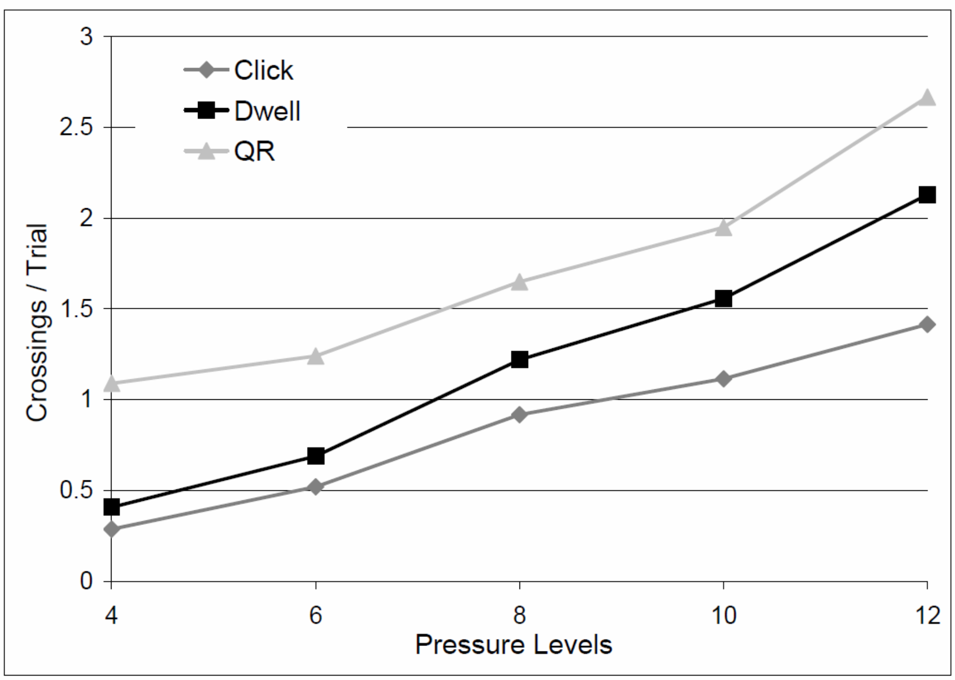


Figure 4.9: Mean number of crossing per trial, by uni-pressure selection mechanism and number of pressure levels.

4.3.3 Subjective Ranking

In the exit questionnaire we asked participants to rank the different selection techniques and sensor locations in terms of preference. Most participants preferred *click* (6 first place rankings, 3 second place rankings) followed by *dwell* (2 first place rankings, 4 second place rankings and 3 third place rankings) and *quick release* (1 first place ranking, 2 second place rankings and 6 third place rankings). Most participants preferred the left location for controlling the pressure sensor (6 first place rankings, 3 third place rankings) followed by top (3 first place rankings, 5 second place rankings and 1 third place ranking) and then right (4 second place rankings and 5 third place rankings). We also asked participants to rank the different selection techniques for each sensor location. The results were similar to the overall preference of the selection techniques.

4.3.4 Discussion

Here we summarize our findings and discuss the implications of our results for the design of pressure-augmented mice.

Selection Mechanism

The results of our first study show that participants were fastest, had a higher level of control (as indicated by the number of crossings) and highly preferred the *click* selection technique. This result is different from that reported by Ramos and colleagues [55] for the tablet PC stylus, in which they found performance with QuickRelease to be the fastest. There are several possible reasons for this discrepancy. With a stylus, it is difficult to click the barrel button while maintaining a consistent level of pressure. With a mouse, clicking a button is a familiar and traditional form of command input. Furthermore, quickly lifting a single finger from the surface of a mouse is less natural that quickly lifting a pen from the tablet PC surface.

Our results indicate that *dwell* is a good selection mechanism as seen by the lower number of errors (0.01 average errors per trial). This result agrees with the results reported by Ramos and colleagues [55]. One explanation for this is that with *dwell* users can ensure the correct object is selected by dwelling on it for a sufficiently long period of time. However, users seem to have difficulty maintaining a consistent level of pressure for an extended period of time; this is evidenced by the high number of crossings with *dwell*, particularly when the number of discrete pressure levels is large. Additionally, our study used a dwell trigger with a 1 second delay; it is possible that with a smaller delay users would perform equally well with *dwell* as they do with *click*. However, smaller delays may also result in a larger number of errors.

Although the *click* technique was fastest and had the fewest number of crossings, error rates were relatively high (0.26 errors per trial). However, when the number of pressure levels was 4 the *click* technique had only 0.11 errors per trial. This result is similar to results found in other research [55]. While this is a relatively large number of errors, it may be possible to reduce this number of errors using more sophisticated pressure distribution functions [62].

Sensor Location

We found that participants were significantly slower with the right sensor location and preferred it the least of all the locations. Our results do not favour the design choice of mounting pressure sensors for the third (ring) and fourth (little) fingers to use.

Pressure Levels

Results of trial completion time, number of crossings and errors indicate that performance degrades rapidly when there are more than six pressure levels. This result is supported by prior studies with pen-based interfaces that suggest it is difficult to control more than six pressure levels [49, 55]. In experiment two, we extend the design of the uni-pressure augmented mouse by affixing an additional pressure sensor to determine if this limit can be extended.

4.4 Design of a Dual-Pressure Mouse

Augmenting the mouse with one pressure sensor provides additional expressive control to users, but limits the number of accessible pressure levels to approximately six. This is a relatively small number of discrete selections, and many possible interactions could benefit if a larger number of pressure levels were selectable. Additionally, a uni-pressure augmented mouse does not facilitate bidirectional input – that is, pressure input that can be interpreted in both positive and negative directions. These limitations can be addressed with dual-pressure control mechanisms.

4.4.1 Dual-Pressure Control Mechanisms

The dual-pressure augmented mouse uses one pressure sensor that is controlled by the second (middle) finger and the other controlled by the thumb. Here we present two dual-pressure control mechanisms: *switch-to-refine* and *tap-and-refine*.

Switch-to-Refine

Switch-to-refine allows users to switch between two sensors to control a large range of pressure values. The range of pressure values are divided such that users select from a coarse-level group of pressure values first, then from the fine-level items in the course-level group (Figure 4.9). In this pressure control mechanism the participant uses one sensor (the primary sensor) to jump through the coarse-level groups and switches to the other sensor (the secondary sensor) to control and navigate in a fine manner through the values in the coarse-level group. To assist the user, the primary sensor does not respond while the user is refining their selection with the secondary sensor. Once the user reaches the appropriate pressure level they click on the left mouse button to select the item. By dividing a large number of pressure levels into course-level groups, each sensor still controls only a small number of levels, but the user is able to select from a large number of targets.

For example, if the total number of selectable items is 36, we can group the items into 6 coarse-level groups each containing 6 fine-level items (Figure 4.9). To select the 11th item, the user applies pressure to the primary sensor until they have highlighted the 2nd coarse-level group (items 7 to 12 in the entire range). Then, the user switches to the secondary sensor to navigate through each of the fine-level items in coarse-level group 2 – that is, from items 7 to 12. To select the 11th item the user applies 5 levels of pressure with the secondary sensor. This technique allows users to select $n \times m$ levels where n and m are the maximum number of pressure values that users can control with the primary and secondary sensors, respectively. Unfortunately, switching from one sensor to the next creates additional overhead in *switch-to-refine*.

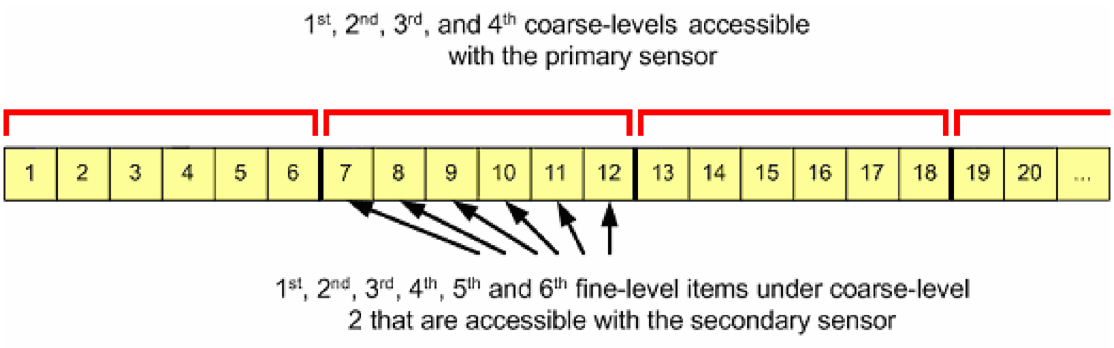


Figure 4.10: An example categorization of targets into coarse-level and fine-level items for the *switch-to-refine* and *tap-and-refine* techniques.

Tap-and-Refine

Tap-and-refine categorizes pressure values into coarse-level and fine-level items similar to *switch-to-refine*. However, the interaction method for selecting the coarse-level targets is different. The user iterates through the coarse-level groups by tapping (by performing a quick press and release within 100ms) the sensors. Each sensor is mapped to a different direction, so tapping the top sensor moves up through the course-level groups and tapping the left sensor moves down through the course-level groups. Once the desired coarse-level is highlighted, the user accesses the finer levels by applying pressure to either of the pressure sensors.

For example, to select the 11th item, the user taps the thumb sensor once. Then the user applies pressure to either sensor until the 11th item is highlighted, clicking the left mouse button to select the item. Tapping each sensor allows the user to move through groups bidirectionally. As a result, users can easily adjust any overshoots that result from tapping too quickly.

4.5 Dual-Pressure User Study

Continuing our experiments to identify the most effective design parameters for pressure-augmented mice we evaluated the dual-pressure control mechanisms we designed and investigated the benefits and trade-offs of uni-pressure and dual-pressure input.

4.5.1 Methods

The goal of this experiment was to determine if dual-pressure control mechanisms can provide improved performance in terms of the number of selectable pressure levels. The experiment was also designed to examine performance differences between the two dual-pressure mechanisms, and the best uni-pressure mechanism from the first study. The experimental task and stimuli were nearly identical to our first study, except that a course-level highlight was added for the *tap-and-refine* and *switch-to-refine* mechanisms (Figure 4.10).

Participants and Apparatus

Eight participants (seven males and one female) between the ages of 21 and 26 were recruited from the University of Saskatchewan for this experiment. All participants were students, all had previous experience with graphical interfaces and all used the mouse in their right hand. The apparatus was identical to that of our first study, except that we used a pressure-augmented mouse with two sensors, one in the top location and one in the left location.

Procedure and Design

The experimental task and the performance measures collected were the same as for the previous experiment. The study used a 4×3×4 within-participants factorial design. The factors were:

- Pressure Levels: 4, 12, 16, 64.
- Control Mechanism: Switch-to-Refine, Tap-and-Refine, Normal.
- Target Distance: 395, 535, 675, 815.

The order of presentation controlled for *control mechanism*. Levels of the other two factors were presented randomly. At the beginning of the experiment we explained the selection mechanisms and participants were given ample time to practice.

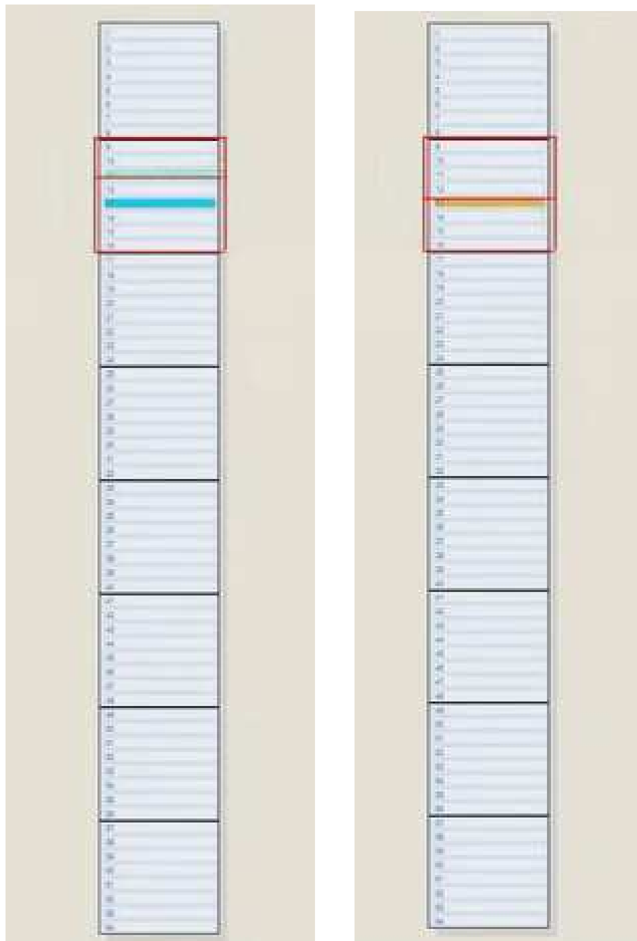


Figure 4.11: Visual examples of the second study's stimuli. (*Left*) The user moved the red group highlight and the red pressure cursor to the blue target. (*Right*) When the target was selected correctly, the target's color changed from blue to yellow.

Pilot trials showed that users were unable to control 64 levels of pressure with the *normal* technique. As a result, the *normal* technique was only tested for pressure levels 4, 12 and 16.

The experiment consisted of three blocks each with five repetitions per condition. With eight participants, four pressure levels for the *switch-to-refine* mechanism, four pressure levels for the *tap-and-refine* mechanism, three pressure levels for the *normal* mechanism, four target distances, three blocks, and five repetitions, the system recorded a total of 5280 trials per participant. The experiment took approximately 60 minutes per participant.

Pressure Control Mechanism

We evaluated *switch-to-refine* and *tap-and-refine* and compared these to the *normal* mechanism. The *normal* mechanism was the uni-pressure *click* mechanism from the first experiment, but allowed participants to choose either the left or top sensor at their preference. All control mechanisms used the *click* selection mechanism from experiment one to confirm target selection.

4.6 Results

Here we discuss in detail the findings of our dual-pressure augmented mouse study, organized by dependent variable (trial completion time, crossings and errors).

4.6.1 Trial Completion Time

The overall mean completion time across all conditions was 1.57s (standard error = 0.044s). We used the univariate ANOVA test of trial completion time and Tamhane post-hoc pair-wise tests (unequal variances) for all our analyses. To make the data conform to the homogeneity requirements for ANOVA we used a natural-log transform on trial completion time. Results show a main effect of c*ontrol mechanism* ($F_{2,14}$=18.46, p<0.01) and *pressure levels* ($F_{3,21}$=178.106, p<0.01) on trial completion time.

Post-hoc pair-wise comparisons of *pressure levels* yielded significant differences (all p<0.01) in trial completion times for all pairs except between 12 pressure levels and 16. Users were fastest with 4 pressure levels, followed by 12, 16 and 64.

Post-hoc pair-wise comparisons of *control mechanism* yielded significant differences (all p<0.01) in trial completion time between Tap-and-Refine and Normal, and Tap-and-Refine and Switch-to-Refine. We did not find a significant difference in trial completion time between Normal and Switch-to-Refine ($p > 0.05$). Users were fastest with Tap-and-Refine followed by Normal and Switch-to-Refine. Figure 4.11 shows the mean trial completion time of each technique by pressure level.

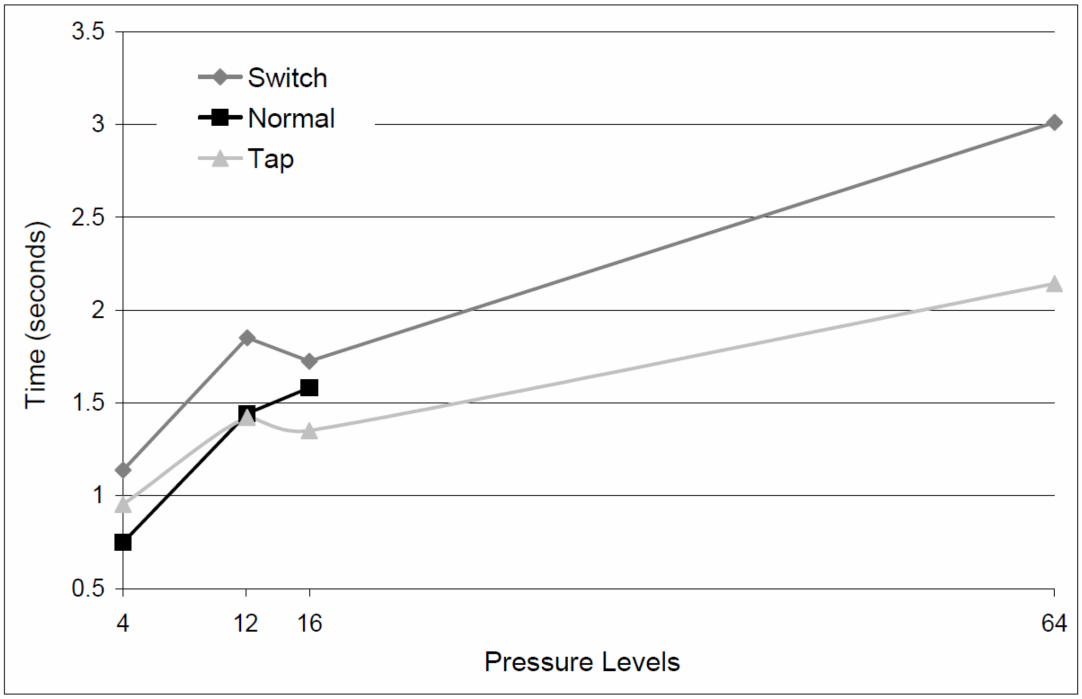


Figure 4.12: Mean trial completion time, by dual-pressure selection mechanism and number of pressure levels. Note that the y-axis starts at 0.5s to better highlight differences.

4.6.2 Crossings and Errors

A univariate ANOVA test revealed a significant effect of c*ontrol mechanism* on number of crossings ($F_{2,14}$=19.101, $p<0.001$). Post-hoc pair-wise comparisons of *control mechanism* showed that Tap-and-Refine had significantly (all $p<0.001$) fewer crossings than all other techniques.

A univariate ANOVA test revealed that a significant effect of p*ressure levels* on number of crossings ($F_{3,21}$=39.764, $p<0.001$). Post-hoc pair-wise comparisons show that *pressure level* 4 had significantly fewer crossings than 12, 16 and 64. However, we found no significant difference in crossings between the other levels. Figure 4.12 shows the average number of crossings for each pressure level.

The average number of errors across all conditions was 0.25 errors per trial (standard error = 0.01). Tap-and-Refine and Switch-to-Refine had 0.17 errors per trial followed by Normal (0.47 errors per trial). The ordering of average number of errors per trial for different *pressure levels* was 4 (0.12), 64 (0.25), 12 (0.31), and 16 (0.32).

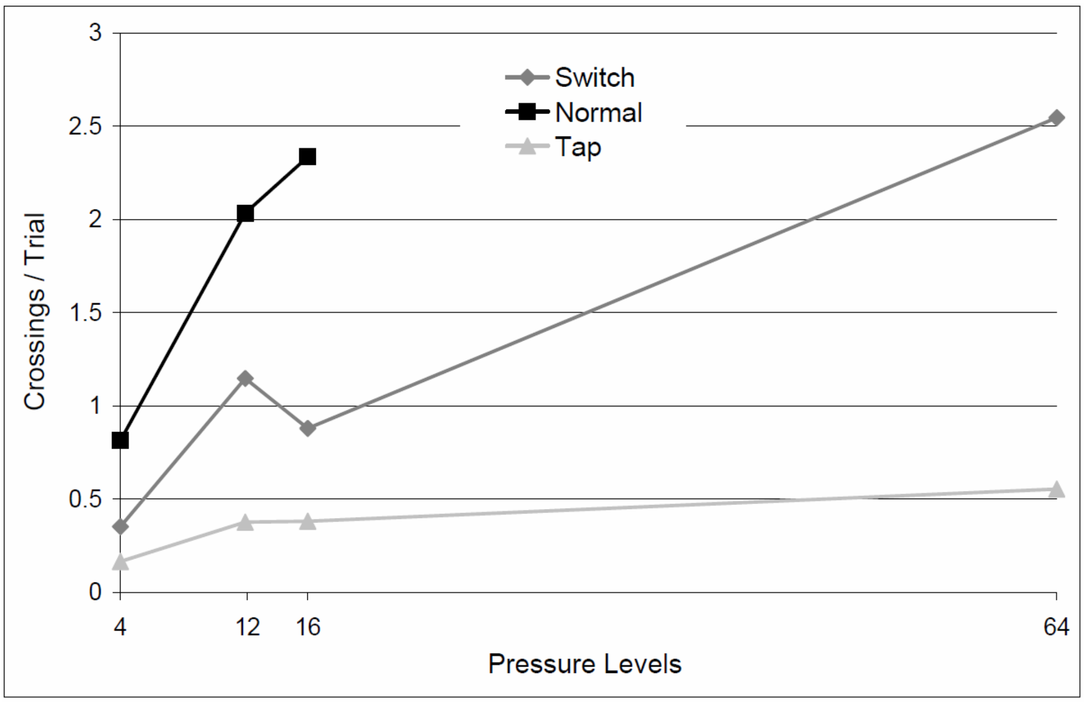


Figure 4.13: Mean number of crossings per trail, by dual-pressure selection mechanism and number of pressure levels.

4.6.3 Discussion

The results of the second study show that the mouse can be augmented with more than one pressure sensor to extend the user's pressure control range. In the following sections we discuss the benefits and limitations of the various pressure control mechanisms we developed, identify application areas that can benefit from a pressure-augmented mouse, and summarize the main lessons for practitioners.

Uni-Pressure Control Strategies

The experimental software recorded continuous time and raw pressure values during each trial. A typical trace of a user's selection task when using the *click* mechanism is shown in Figure 4.13. A user's application of pressure can be characterized by two steps: a coarse-grained impulse of pressure input to get close to the target and then fine-grained precision movement to select the target. In the coarse-grained movement users quickly apply an amount of pressure to get in the range of the desired pressure value. Then, more slowly, users apply or release pressure to select the appropriate target.

More precisely, we notice that once users get within the vicinity of the target they take between 150 and 300 ms to refine their pressure movement to select the target. This is often the time it takes the user to feel confident that they have the correct pressure value and momentarily switch their attention to their first finger for clicking the mouse button. Further enhancements to the interactions, particularly in this refinement stage, could improve performance with the *click* selection mechanism and possibly improve error rates [62].

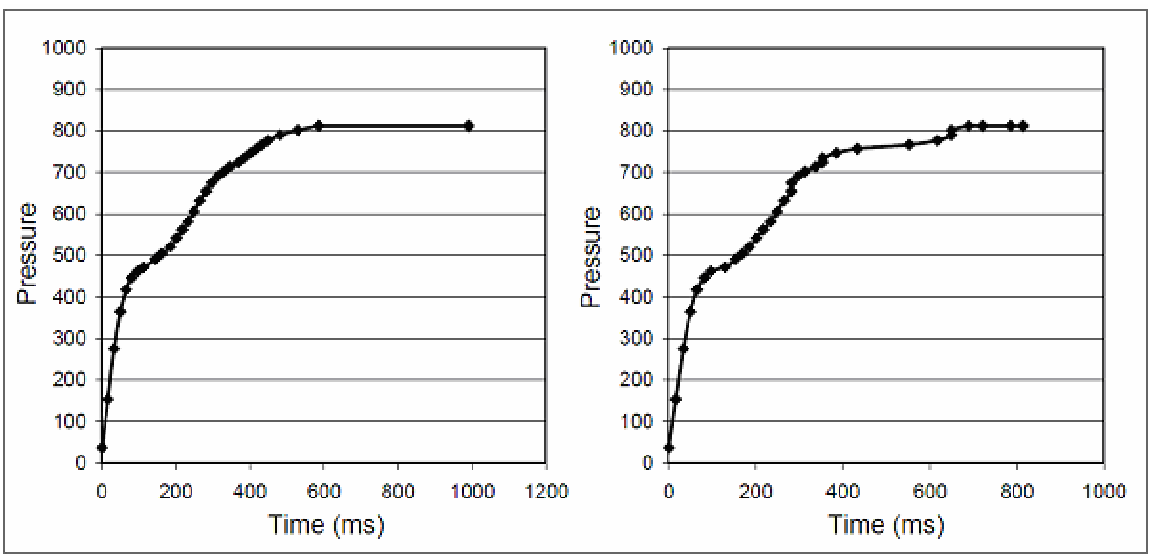


Figure 4.14: Two traces of user pressure data when performing a uni-pressure selection task with the *click* selection mechanism.

Dual-Pressure Control Strategies

Users were faster and better able to control pressure with the *tap-and-refine* technique than with the *switch-to-refine* technique. In our study we compared the two techniques at 64 discrete levels. These were separated into 8x8 discrete levels. As a result, adding more levels to any of the two sensors would result in significant performance decreases, particularly with *switch-to-refine*.

While it is possible that the tapping action in *tap-and-refine* could be replaced by a traditional button, such a design would require two additional buttons (one for each direction) and one pressure sensor to work effectively. However, analysis of our log files suggest that typical tap times are between 50 and 80 ms, which may be faster than button click times [41]. We further investigate the interchangeability of button clicking and pressure sensor tapping in Chapter Five.

4.7 Summary

In this chapter we answered a number of key design questions for pressure-augmented mice. Results of our first study show that pressure buttons are best controlled with the second (middle) finger and the thumb. Our first study also confirmed that users can control six pressure levels with one pressure button, and that selection tasks are best confirmed with a button click. We were also able to determine that the best pressure discretization function was quadratic centered at the lowest pressure level. The results of our second experiment showed that with two sensors and the *tap-and-refine* mechanism users can select from 64 targets. *Tap-and-refine* can also provide pressure input in a bidirectional manner.

There are several lessons that designers can take from our experiments:

- Place pressure buttons so that they are accessible by the middle finger and the thumb.
- Limit the number of pressure levels selectable with a single pressure sensor to six.
- Use mouse button clicks for confirming pressure-based discrete target selections.

- Use dual-pressure mechanisms for increasing the selectable range of pressure levels and modes.
- *Tap-and-refine* is the most effective control mechanism for providing bidirectional pressure input and controlling a large number of pressure levels.

With the results of this study of the first pressure-augmented mouse configuration in mind, we continued our research to augment the mouse with pressure-sensitive input. In the following chapter, we test the second pressure-augmented mouse configuration, where mouse buttons are replaced with pressure sensors.

CHAPTER 5

DESIGN GUIDELINES FOR PRESSURE BUTTONS

In this chapter we continue our research to provide ergonomics and performance information for pressure-augmented mice by testing a second pressure-augmented mouse configuration. While the designs discussed in the previous chapter perform well, a design that improves the capabilities of the mouse without requiring additional buttons is potentially more compelling; by improving the capabilities of the existing left and right mouse buttons, it may be possible to offer pressure control directly under the user's finger tips in locations they are familiar using, also avoiding any possible usability issues that may arise from installing multiple additional buttons on the mouse. Such a design would seamlessly integrate pressure interaction into the mouse by substituting or augmenting the left and right mouse buttons with pressure sensors (Figure 5.1).

It should be noted that while our first configuration avoids interfering with standard clicking, this second configuration attempts to make the mouse more expressive by replacing the standard mouse buttons with pressure sensors. Therefore, this second configuration requires pressure input from the first (index) finger, which is not the case in the first configuration. For this second configuration to be successful a number of our key design questions must be answered; the most effective means for simulating clicks and double-clicks with pressure sensors and feedback requirements must be determined.

Figure 5.1: (*Left*) The left mouse button is disabled by taping it down and a pressure sensor is affixed and used as a pressure button. (*Right*) A pressure sensor is installed underneath the left mouse button casing creating a pressure-augmented mouse button.

To facilitate an interchangeability of pressure sensors with mouse buttons, at a minimum two primary and fundamental actions performed with standard mouse buttons would need to be replicated: selection and action invocation. Selection is commonly performed with a single-click, while triggering an action (such as opening a file or application) is typically performed with a double-click. Pressure input may also allow for multiple levels of a single-click, similar to the design proposed by Zeleznik and colleagues [70] in pop-through mouse buttons; however, pop-through buttons only support three discrete states (soft click, hard click, release) while a pressure sensor is capable of supporting as many as six.

In this chapter we set out to provide ergonomics and performance information for our second mouse configuration. We introduce and evaluate several interaction designs for selection and action invocation with pressure sensors in two main studies and a third informal study. In our first study we tested four pressure-based selection techniques and compared them to a standard mouse button click. Results show that it may be possible to effectively replace button click actions with *pressure tap* actions. In our second study we tested four pressure-based action invocation techniques and compared them to a standard mouse button double-click. Results show that a hard press action with a pressure sensor is faster than a standard mouse button double-click. Subjective results from both the first and second study indicate that users may prefer standard mouse buttons, perhaps due to the familiar haptic feedback provided by standard buttons. In a third study we tested a standard mouse button augmented with pressure-sensitivity (Figure 5.1 *right*). In this study, the hard press double-click technique is compared to a standard mouse button double-click. Preliminary results suggest that pressure-augmented mouse buttons may offer the benefits of both pressure input and standard buttons.

The main contributions of this chapter are:

1. Identification of the most effective design parameters for a pressure-button mouse, including selection and action invocation techniques and feedback requirements for single and double-click tasks.

2. Identification of possible drag-and-drop techniques for pressure buttons.

3. A set of design recommendations for future pressure-button mice.

5.1 Design of Selection and Action Techniques

In order for this second pressure-augmented mouse configuration to be successful, pressure buttons must effectively simulate the primary operations of selection and action invocation; in other words, clicking and double-clicking. A key point that distinguishes pressure buttons from current mouse buttons is that current button clicks are based on users clicking past a predetermined pressure value that is essentially hardwired in the design of the button. Pressure-based selection techniques offer other alternatives to this; clicks can be activated using low pressure thresholds and timeouts, and double-clicks by pressing harder.

Most of the design factors for pressure-augmented mice that were presented in Chapter Four still apply to this configuration. However, because the task is different, different selection techniques must be designed, and the number of required pressure levels is small. The nature of this second mouse configuration also means that the sensor locations are fixed. In the following section we discuss the relevant pressure-augmented mouse design factors for selection and action invocation tasks.

5.1.1 Selection Techniques with Pressure Buttons

We designed four pressure selection techniques designed to emulate clicking actions with pressure sensors. Two of the techniques include audio feedback; Akamatsu and colleagues [2] and MacKenzie and Oniszczak [44] suggest that aural feedback is essential to the closed-loop feedback of clicking on a mouse button.

- *Pressure Tap*: In this selection technique a click is registered if the user applies and releases pressure within a time interval of T. *Pressure tap* was inspired by the success of the dual-pressure *tap-and-refine* mechanism from Chapter Four. Figure 5.2 shows the invocation of a mouse click with *pressure tap*. The entire click-and-release operation is essentially one atomic unit. The click is triggered when the user applies and then releases a small amount of pressure (a threshold value of 2 units from the pressure sensor and Phidgets interface kit [22]) within 150 milliseconds. If the user does not apply pressure beyond the threshold and then release within the specified time interval, the system does not register a click.

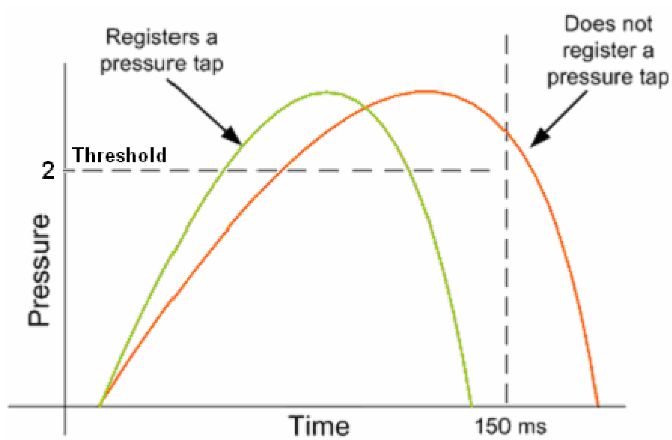


Figure 5.2: With *pressure tap* a mouse click is invoked by applying pressure beyond a minimum pressure threshold and releasing pressure within a time interval of T. In our implementation a T of 150 ms and a minimum pressure threshold of 2 units were used.

- *Pressure Tap Audio*: In this form of the *pressure tap* technique a higher pitched click sound is played when pressure is first applied and a lower pitched click sound is played if a click is successfully registered.

- *Pressure Click*: This selection technique is designed to replicate the operation of a mouse button click as closely as possible. Applying a pressure of P_{down} invoked a mouse down event. After triggering a mouse down event a mouse up event was triggered when the pressure level become lower than P_{up}. A pressure-timing graph in Figure 5.3 depicts the invocation of a mouse down and mouse up with a pressure sensor and a button.

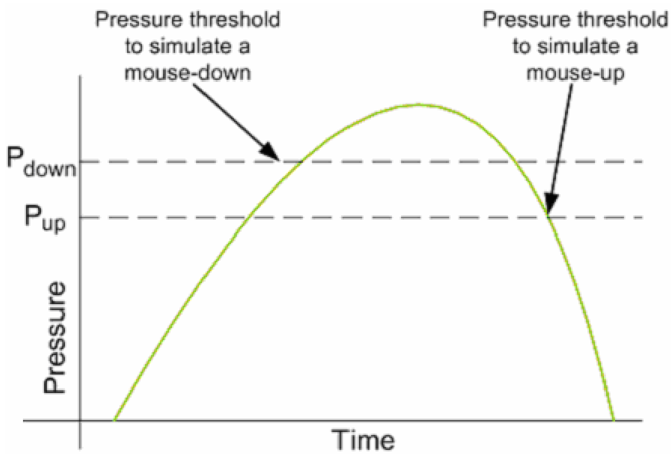


Figure 5.3: With *pressure click* a mouse down is invoked by applying pressure beyond a threshold of P_{down}. A mouse up is invoked by releasing pressure below a threshold of P_{up}. In our implementation a P_{down} of 4 pressure units and a P_{up} of 2 pressure units were used.

- *Pressure Click Audio*: In this form of the *pressure click* technique a higher pitched click sound is played when the pressure level reached P_{down} and a lower pitched click sound played when the pressure level reached P_{up}.

5.1.2 Action Techniques with Pressure Buttons

Users typically invoke actions such as opening a file or executing a program by double-clicking the mouse button. We designed four action invocation techniques:

- *Pressure Double-Tap*: With pressure double-tap a double-click is triggered by performing two *pressure tap* actions in series. The time delay between the two taps is similar to the delay required to register a double-click using a mouse button. In most systems this delay is configurable to match the users' motor capacities**.** However, as with *pressure tap*, each tap action must be completed within a fixed time interval of T.

- *Pressure Double-Click*: With pressure double-click a double-click is triggered by performing two pressure clicks in series. The time delay between the two pressure clicks is similar to the delay required to register a double-click using a mouse button. In most systems this delay is configurable to match the users' motor capacities**.**

- *Hard Press*: With hard press a double-click is triggered when the user applies pressure beyond a set pressure threshold. In our experiments this threshold was configured individually for each participant. Figure 5.4 depicts the hard press and pressure double-click techniques in a pressure-time graph.

- *Hard Press Audio*: In this form of the hard press technique a double-click sound is played when a successful hard press is performed.

5.1.3 Visual Feedback

Based on guidelines from Li and colleagues [41], Mizobuchi and colleagues [49], and Ramos and colleagues [55] feedback is a necessary component for the proper functioning of pressure input. Unlike mouse buttons, pressure sensors do not provide any aural or tactile feedback upon being pressed or released; this lack of feedback could adversely affect performance with pressure buttons, especially the techniques without audio feedback.

In our experimental setup we provided visual feedback to show when the user had invoked mouse down and mouse up events. However, unlike the pressure control tasks studied in Chapter Four which required continuous visual feedback for accuracy, pressure clicking relies on rapidly applying and releasing pressure. As a result, pressure buttons cannot harness any additional benefits from continuous visual feedback. Visual feedback with pressure buttons was provided by highlighting the target in orange when the sensor was pressed down and in green when released.

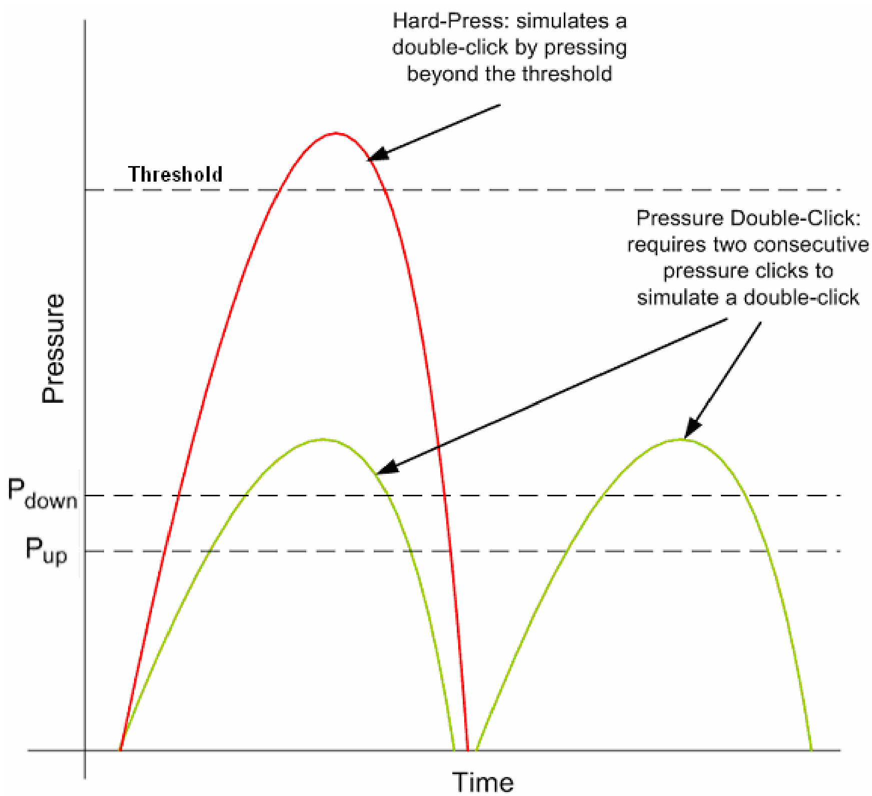


Figure 5.4: With *hard press* a double-click is activated by applying pressure beyond a threshold that is significantly higher than the *pressure click* single-click threshold. With *pressure click* two consecutive pressure clicks are required to enter a double-click.

5.2 Selection User Study

To provide design information for the pressure button mouse configuration, we performed two main studies. This first study was designed to determine the most effective pressure-based selection technique and the auditory feedback requirements for pressure-based selection. Our second experiment focused on pressure-based action invocation.

5.2.1 Hardware Configuration

Our study used an optical mouse with pressure sensors mounted onto the surface of the mouse buttons (Figure 5.1). In the pressure-based selection conditions the mouse buttons were disabled by taping them down so that they could no longer be activated by pressing on the sensors. In the standard mouse click condition the pressure sensors were removed from the mouse. The sensor (model #IESF-R-5L from CUI Inc.) could measure a maximum pressure value of 1.5 N. Each sensor provided 1000 pressure levels through the Phidgets [22] interface kit. The application was developed in C# and the experiment was conducted in full-screen mode at 1024×768 pixels on a P4 3.0 GHz Windows XP OS machine.

A known limitation of our apparatus is that in comparison to the regular mouse buttons, the pressure sensors covered a very minimal area or *footprint* (see Figure 5.1). This means that users may make contact with the pressure sensor with the side of their first or second finger when clicking, causing an intended click or tap to go unregistered. As a result, users could potentially take longer to trigger a selection with pressure buttons. Pressure sensors with a larger sensing area could alleviate this impediment.

5.2.2 Performance Measures

The experimental software recorded trial completion time and errors as dependent variables. Trial completion time is defined as the total time taken from the presentation of visual stimulus indicating the beginning of the task to the correct completion of the task. The software records an error when the participant performed an action but did not complete the task. For example, in the *pressure click* technique errors occur when the user does not press the pressure sensor hard enough for the system to register a mouse down event. The trial ended only when the user completed the task, so multiple errors were possible for each trial. Participants were also asked in an exit questionnaire to rank the different selection techniques.

5.2.3 Methods

The main goal of this experiment was to determine if pressure-based selection techniques perform as well as standard mouse button clicking. The experiment was designed so that the four pressure selection techniques could be compared with standard mouse button clicking, and so that techniques with audio feedback could be compared to the standard form of the techniques.

Participants

Ten participants (four males and six females) between the ages of 19 and 25 were recruited from the University of British Columbia. All participants were students, all had previous experience with graphical interfaces and all used the mouse in their right hand.

Tasks and Stimuli

We used a simple object selection task in which the participants were asked to perform a single-click action when the target object turned green. At the beginning of each trial a timer counts down to zero. When the target object changes color from white to green, the user performs the single-click selection action. A three second countdown timer was used to cue the user so that they were prepared to perform the action as quickly as possible. Since we are primarily interested in recording motor-response times we felt this was an effective way to control reaction times and errors due to unpreparedness.

The user did not have to move the cursor to perform their task and they were instructed to avoid moving the cursor. However cursor movement was not disabled to maintain a task that would be more ecologically valid. During each trial the user performed the selection action using the different selection mechanisms according to the predefined order of presentation.

Procedure and Design

The study used a 2x5 within-participants factorial design. The factors were:

- *Location*: Left (first finger), Right (second finger).
- *Selection Technique*: Button Click, Pressure Click, Pressure Click Audio, Pressure Tap, Pressure Tap Audio (see section 5.1.1).

The order of presentation first controlled for *location* and then for *selection technique*. Before the study we explained each of the selection techniques to the participant. Each participant was given an abbreviated version of the full experiment for training, which lasted approximately 10 minutes. The experiment consisted of three blocks with each block consisting of twenty repetitions for each condition. With ten participants, two device locations, five selection techniques, one task, three blocks, and twenty trials, the system recorded a total of (10×2×5×1×3×20) 6000 trials. The experiment took approximately 40 minutes per participant.

Location

For the purposes of testing this mouse configuration, we are testing two device locations: the left mouse button location, usually controlled with the first (index) finger, and the right mouse button location, usually controlled with the second (middle) finger. These locations are important because they are the locations of the majority of mouse button usage.

Selection Technique

The selection techniques we tested include the four pressure-based selection techniques discussed in section 5.1.1, as well as standard button clicking with a typical two button mouse.

- *Button Click*: Button click consisted of a single click with the mouse button.
- *Pressure Click*: With *pressure click* and *pressure click audio* we use a P_{down} of 4 pressure units and a P_{up} of 2 pressure units (see section 5.1.1). In the *pressure click audio* condition, users heard a click sound when each of the P_{down} and P_{up} levels were crossed.
- *Pressure Tap*: With *pressure tap* and *pressure tap audio* we use a time interval T of 150ms and a minimum pressure threshold of 2 units. In the *pressure tap audio* condition, users heard a click sound when the pressure tap was activated successfully.

5.3 Results

Here we discuss the results of our first study comparing pressure-based selection techniques to standard mouse button clicking, organized by dependent variable (completion time, errors and subjective feedback).

5.3.1 Completion Time

We used the univariate ANOVA test of completion time for our analyses. To make the data conform to the homogeneity requirements for ANOVA we used a natural-log transform on the completion time and only included in our analysis trials that were successfully completed. Results showed no main effects of *selection technique* ($F_{4,26}$= 2.551, p>0.05) or *location* ($F_{1,6}$=1.198, p>0.05) on trial completion time. Users were on average faster when selecting with the right location. Users were fastest with Button Click followed by Pressure Tap, Pressure Tap Audio, Pressure Click Audio and Pressure Click (Figure 5.5).

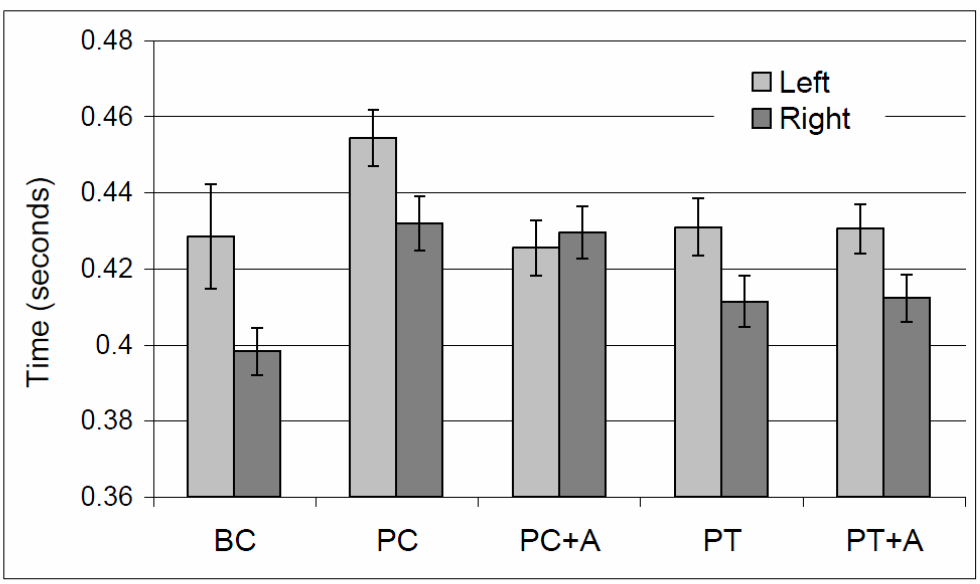


Figure 5.5: Mean completion time, by selection technique and location. Standard error bars are also included. Note that the y-axis begins at 0.36 seconds to better highlight differences. (BC: Button Click, PC: Pressure Click, PC+A: Pressure Click Audio, PT: Pressure Tap, PT+A: Pressure Tap Audio.)

5.3.2 Errors and Subjective Feedback

Across all conditions the total number of errors was 272 (0.06 errors per trial). The distribution of errors is as shown in Figure 5.6. Errors were not recorded for the Button Click condition. In terms of overall effort required, seven of the ten subjects preferred Button Click while three preferred Pressure Click Audio.

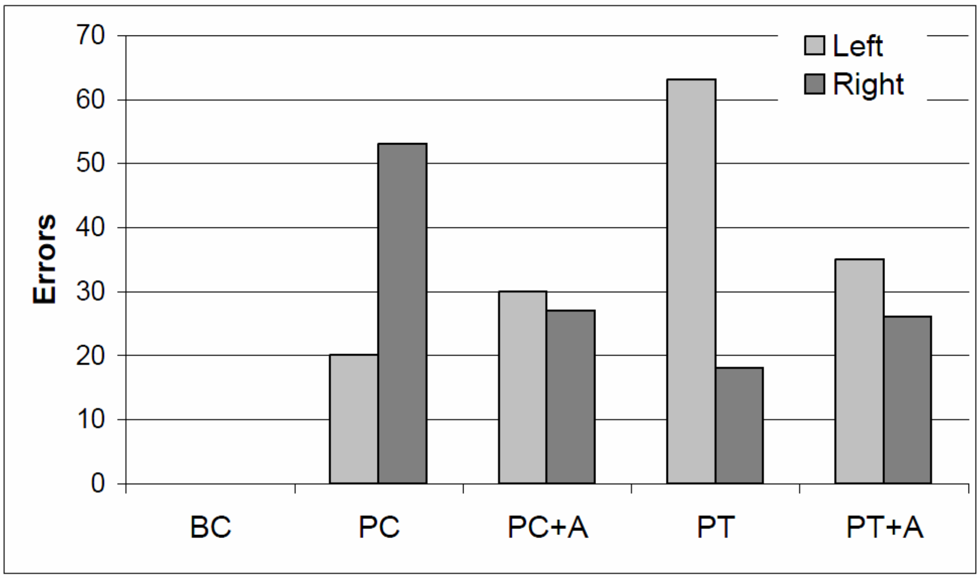


Figure 5.6: Number of errors, by selection technique and location. (BC: Button Click, PC: Pressure Click, PC+A: Pressure Click Audio, PT: Pressure Tap, PT+A: Pressure Tap Audio.)

5.3.3 Discussion

We did not detect any significant difference in completion time between the selection techniques. Our results indicate that users were fastest with Button Click. The average difference in completion time between Button Click and Pressure Tap was less than 80 milliseconds. As shown in Figure 5.7, even though users press relatively hard on the sensors (users' peak pressure value ranged from 100 to 200 pressure units), users could quickly engage and disengage interaction with the pressure sensor.

As mentioned earlier, the footprint of the pressure sensor used in the study was smaller than the footprint of the mouse buttons. This difference did not appear to significantly affect completion times in correctly completed trials. However, the size of the sensor footprint did seem to affect the number of errors. Between trials users often rested their finger on the sensor, but when performing the selection action users would often lift their finger and then reacquire the sensor. This behaviour caused errors as users would sometimes not click the central sensing area of the pressure sensor and therefore the system would not detect an intended click or tap. Where possible the experimenter noted these errors manually, finding that 78 (29%) of the total number of errors resulted from users missing the central sensing area of the sensor (Figure 5.6). Since these types of errors were noted manually during the experiment, it is possible that some were missed, meaning that the actual percentage of errors caused by sensor footprint may be higher. If the pressure buttons were designed with a larger footprint we believe the error rate (0.06 errors per trial) would be much lower.

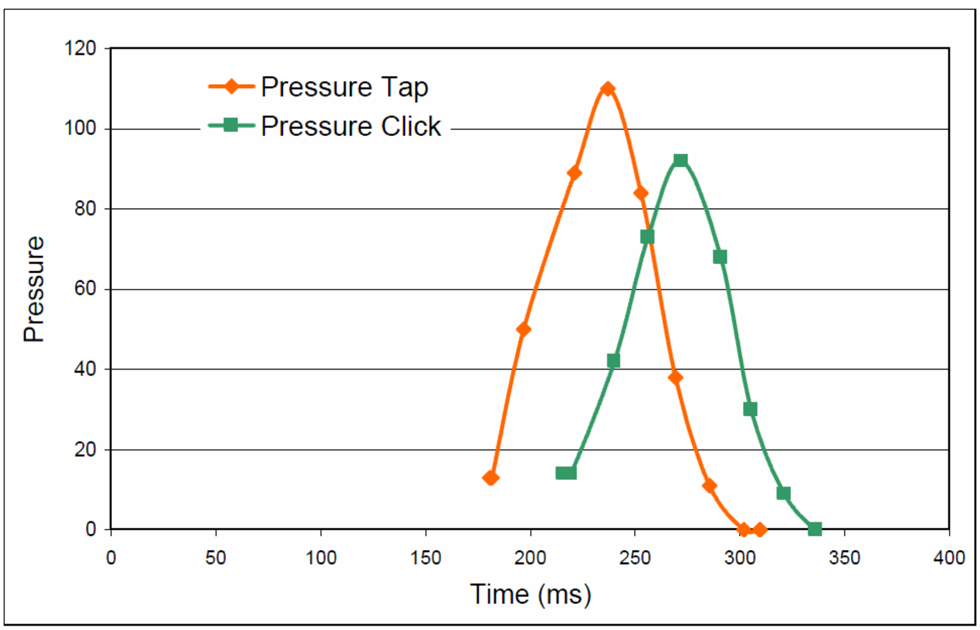


Figure 5.7: Pressure by time plot of two typical single-click actions with *pressure tap* and *pressure click.*

5.4 Action Invocation User Study

The results of the first study identified some drawbacks to the use of pressure sensors as buttons. However, the *pressure tap* technique was not significantly slower than button clicking, and was preferred by some users in spite of the obvious problem of sensor footprint. With these results in mind, we proceed to our study of action invocation with pressure buttons. The main goal of this second study was to determine if pressure-based action invocation techniques are as effective as standard double-click actions.

Participants and Apparatus

Ten participants (five males and five females) between the ages of 19 and 25 were recruited from the University of British Columbia. All participants were students, all had previous experience with graphical interfaces and all used the mouse in their right hand. The experimental apparatus and task were identical to those used in the first study except that users were required to perform a double-click action instead of a single-click action.

Procedure and Design

The study used a within-participants factorial design with *action-invocation technique* as the independent variable**.**

- *Action-Invocation Technique*: Button Click, Pressure Click, Pressure Tap, Hard Press and Hard Press Audio (see section 5.1.2)**.**

The order of presentation controlled for *action-invocation technique*. Before each experiment we explained each of the techniques to the participant. Each participant was given an abbreviated version of the full experiment for training, which lasted approximately five minutes. The experiment consisted of three blocks with each block consisting of twenty repetitions for each condition. With ten participants, five input modes, one task, three blocks, and twenty trials, the system recorded a total of (10×5×1×3×20) 3000 trials. The experiment took approximately 20 minutes per participant**.** As with the previous study, the experimental software recorded trial completion time, and errors as dependent variables. Participants were also asked in an exit questionnaire to rank the different selection techniques.

Action-Invocation Techniques

The techniques we tested included the four pressure-based action invocation techniques discussed in section 5.1.2, as well as double-clicking with a standard two-button mouse.

- *Button Click*: This input mode consisted of the conventional double-click with the mouse button.

- *Pressure Double-Click*: A *pressure double-click* consisted of two consecutive pressure clicks. Pressure values were P_{down} of 4 units and P_{up} of 2 units (see section 5.1.1). No timeout was enforced between the two pressure clicks.

- *Pressure Double-Tap*: A *pressure double-tap* consisted of two consecutive pressure taps (see section 5.1.1). The minimum pressure threshold was 2 pressure units and the *pressure tap* timeout was 150ms. However, no timeout was enforced between the two taps.

- *Hard Press*: A hard press required that users press beyond a certain pressure threshold, but only once. A unique hard press threshold was determined for each participant before the experiment by asking them to perform a series of hard presses and pressure clicks, and noting the recorded pressure values. Hard press thresholds varied between 65 and 185 pressure units. In the Hard Press Audio condition an audible double-click sound was played when the Hard Press threshold was crossed.

Because we did not find any significant difference in terms of completion time between the conditions with and without audio for Pressure Click and Pressure Tap in the first study, we did not include audio feedback enhanced versions of these techniques in this second study, which helped reduce the number of action invocation techniques that participants were presented with.

5.5 Results

Here we discuss the results our action invocation user study testing pressure-based and standard double-click techniques, organized by dependent variable (completion time and errors, and subjective feedback).

5.5.1 Completion Time and Errors

We used the univariate ANOVA test of completion time and Tamhane post-hoc pair-wise tests (unequal variances) for all our analyses. To make the data conform to the homogeneity requirements for ANOVA we used a natural-log transform on the completion time. Results showed a main effect of *action-invocation technique* ($F_{4,29}$=40.19, $p<0.01$) on trial completion time.

Post-hoc pair-wise comparisons of *action-invocation technique* yielded significant differences (all $p<0.01$) in trial completion time for all pairs except Hard Press and Hard Press Audio, and Pressure Click and Pressure Tap. Users were fastest with Hard Press followed by Hard Press Audio, Button Click, Pressure Tap and Pressure Click. Figure 5.8 shows the mean completion time for each mode.

There were a total 147 errors (0.06 errors per trial) across all conditions for all trials. The distribution of errors is as shown in Figure 5.9. Although the experimenter did not note the number of errors caused by missing the central sensing area of the pressure sensor as in the first study, it seems likely that a large number of errors resulted from the small footprint of the pressure sensors.

5.5.2 Subjective Feedback

In terms of overall preference users were split between the techniques; 2 participants preferred Button Click, 2 participants preferred Hard Press, 3 participants preferred Pressure Tap and 3 participants preferred Pressure Click. These results show that most participants preferred Pressure Tap and Pressure Click, which is interesting given that Hard Press and Button Click achieved the best empirical performance. No participants preferred Hard Press Audio.

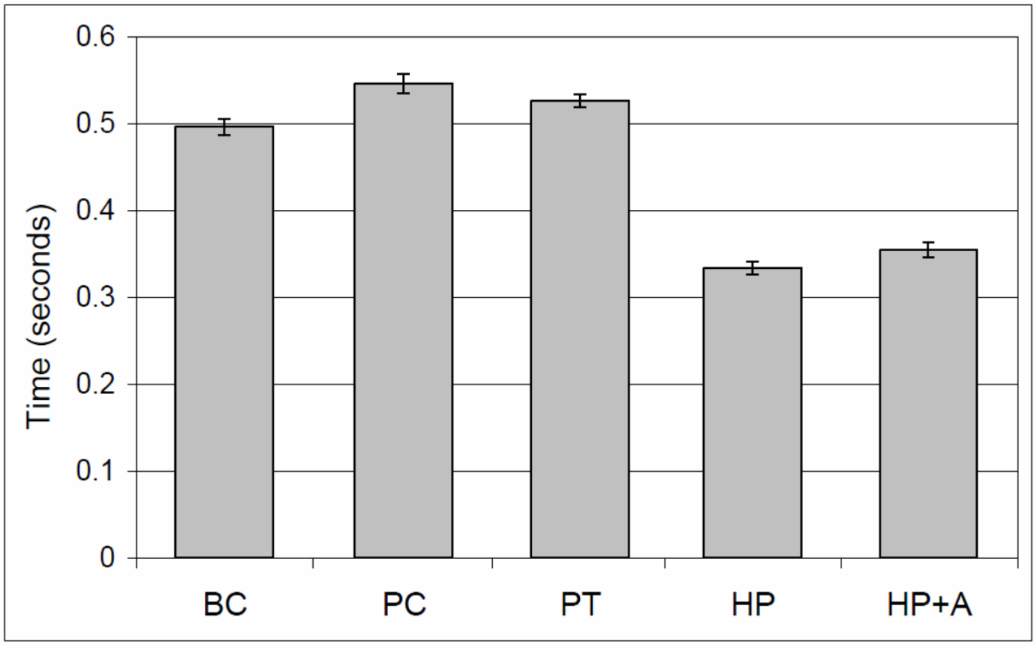


Figure 5.8: Mean completion time, by action-invocation technique. Standard error bars are also included. (BC: Button Click, PC: Pressure Click, PT: Pressure Tap, HP: Hard Press, HP+A: Hard Press Audio.)

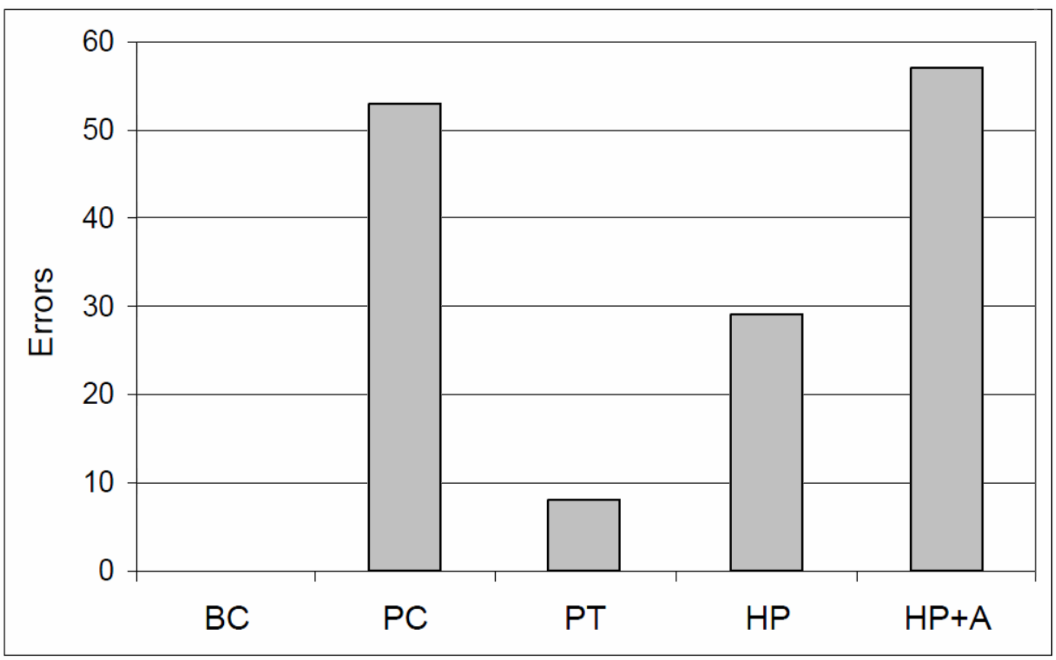


Figure 5.9: Number of errors, by action-invocation technique. (BC: Button Click, PC: Pressure Click, PT: Pressure Tap, HP: Hard Press, HP+A: Hard Press Audio.)

5.6 Summary

To provide design information for pressure-augmented mice with pressure sensors replacing mouse buttons we carried out two user studies; one study testing single-click selections and the other testing double-click action invocations. Despite the small footprint of the pressure sensors, a number of positive results were observed. The results of the first study show that *pressure tap* is not significantly slower than basic buttons for performing selections. In the second study we found that users were significantly faster when using hard press for action invocations. Also, in terms of subjective feedback, *pressure click* performed relatively well in our first study. While our results show that there is promise in both designs, our limited implementations of these devices caused several problems which limit the generalizability of some of our findings. In addition, the high error rate (0.06 errors per trial in both studies) makes pressure buttons, at least in their current form, impractical. A second design iteration, testing pressure sensors with larger sensing areas and other research devoted to reducing error rates is necessary before pressure buttons could be utilized.

In the following sections we discuss in depth the observations we made during both studies. We also present several natural extensions to pressure buttons, the results of an informal study of pressure-augmented mouse buttons, and a list of recommendations for future pressure button designs.

5.6.1 Observations on the Design of Pressure Buttons

The results of our experiments have revealed a number of factors that are particularly important to the design of mice with pressure buttons.

Feedback

The feedback component of any closed-loop interaction is crucial to the proper functioning of an interactive system. In the case of clicking standard mouse buttons, users are given auditory and haptic feedback. However, our results did not show any completion time benefits to auditory feedback. It is also unclear from our results whether or not auditory feedback reduces or increases error rates; while the number of errors is lower for auditory conditions in study one, the number of errors is higher in experiment two. These results are interesting given that pressure buttons do not provide any accurate form of haptic feedback. In terms of visual feedback, previous research with pressure-sensing interactions suggests that continuous visual feedback is necessary for the successful operation of pressure-based input. However, given the short completion times that were observed, selection and action invocation cannot benefit from full continuous visual feedback.

Double-Click Timeouts

An important factor for double click performance with pressure sensors and mouse buttons is the double-click timeout. This timeout that is inherent in double-clicking is important for distinguishing a double-click from two independent single clicks. In most systems users can customize this timeout value to compensate for individual motor capacities. In the case of pressure buttons, *pressure click* and *pressure tap* techniques also rely on a timeout to distinguish a double-click from independent single-clicks. However, in our second study we deliberately did not impose any double-click timeouts for any of the interaction techniques in order to allow for differing motor capacities of participants.

Hard press does not require a timeout to distinguish a double-click from two single-clicks. With hard press, users only need to cross a specified pressure threshold to activate a double-click. However, we noticed that the most appropriate pressure threshold value varied between users. To set the hard press threshold, users performed about 10 practice trials before starting the experiment. We used the values from the practice trials to determine the most appropriate hard press threshold for each user. In our study, hard press thresholds ranged from 65 to 185 pressure units. We envision that in an actual implementation of hard press users will be able to customize the hard press threshold in a manner similar to customizing the double-click timeout.

5.6.2 Natural Extensions to Pressure Buttons

In this section we briefly discuss three possible extensions to our designs that may improve the overall usability of pressure-augmented mice with the pressure-button configuration.

Pressure-Augmented Buttons

There are two main drawbacks to pressure buttons. First, pressure buttons lack haptic ‘clicking’ feedback. While *pressure tap* was not found to be significantly slower than button click for selection actions, users still preferred button click over *pressure tap*. Part of the reason for this may be the lack of haptic feedback from the mouse button. Second, pressure buttons have a small physical footprint that can be difficult to acquire with the finger when clicking. It seems likely that these drawbacks are the cause of the numerous errors recorded for pressure-based selection techniques. However, both of these drawbacks can be removed with pressure-augmented mouse buttons.

Pressure-augmented buttons would incorporate pressure-sensing functionality into the mechanics of a typical mouse button. This new pressure button design would facilitate single clicks with the mouse button, providing familiar haptic feedback from the button. Users can then simulate a double-click using the hard press mechanism by further applying pressure to the pressure-sensing mouse button. This design would give users the flexibility to choose pressure interaction or button-based interaction, and allow a mouse button to be used for standard clicks as well as discrete and continuous pressure control. If such a pressure button design could be implemented then a large number of interactive features can be associated with a single pressure-augmented button, a vast improvement over the pop-through mouse which allows only three states [70].

To identify whether such a concept is possible, we performed a third study. In an informal study of pressure-augmented buttons, we implemented a prototype pressure-augmented mouse button by slipping a model #IESF-R-5L pressure sensor from CUI Inc. underneath the mouse button casing of a Microsoft Comfort Mouse[12]. We tested this new

[12] Microsoft Comfort Mouse
`http://www.microsoft.com/hardware/mouseandkeyboard/productdetails.aspx?pid=072`

pressure-augmented button design by delegating the single-click selection action to the mouse button and double-click action invocations to a hard press interaction. We were also interested in identifying whether performance with a pressure-augmented button would be comparable to performance in our first and second pressure buttons experiments. We carried out this study with ten subjects to examine performance with a pressure-augmented mouse button for both selection and action invocation tasks.

Results from this initial study demonstrate that a pressure-augmented mouse button can facilitate both single-click actions with the mouse button and hard press double-click actions with pressure input. The mean time for button double-click was approximately 250 milliseconds higher than the mean time for button single-click, while the mean time for hard press was only 150 milliseconds higher than the mean time for button single-click. These mean times are comparable to the means that were found in our first and second experiments with pressure buttons. This preliminary result suggests that a professionally designed pressure-augmented button could be just as effective as pressure buttons for hard press, while providing button-based single-click functionality.

From Clicking to Invoking Contextual Pressure Menus

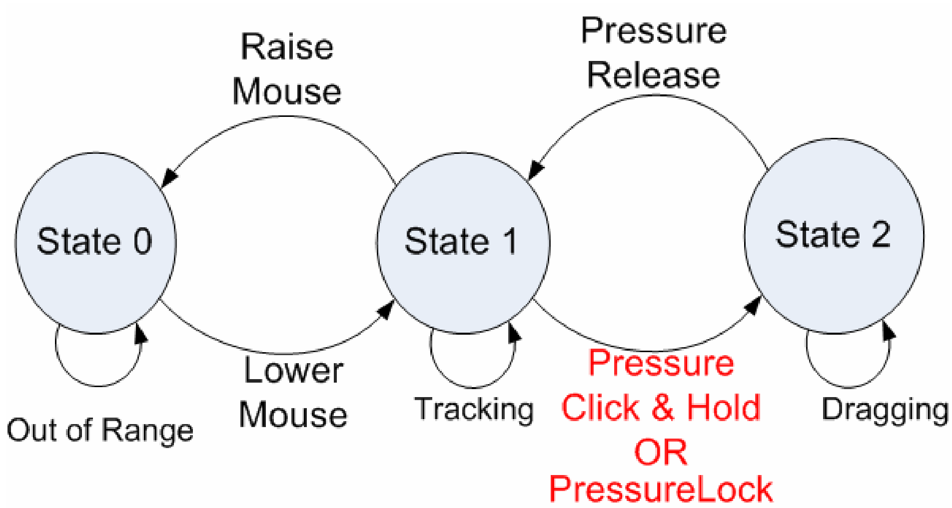
In our second study we did not use an upper pressure threshold for hard press. Such an implementation is perfectly suited to current applications where users only use the mouse buttons for single or double-click actions. However, with pressure buttons it is desirable to also control continuous interactions or discrete interactions with many states, similar to our research in Chapter Four.

To make use of the pressure space provided by a pressure sensor, an implementation of hard press should include an upper pressure threshold. When the user applies pressure beyond this threshold, the system can enter into a continuous or discrete pressure interaction mode; essentially, the first discrete pressure level would correspond to a double-click. In theory, hard press could be combined with interactions similar to those tested in Chapter Four, since the pressure values used in hard press appear in the lower pressure range (65 to 185 pressure units). This allows designers to use the upper range of pressure values (values greater than 185 pressure units) for continuous or discrete pressure interactions. By examining the pressure levels that were used in our research in Chapter Four, up to five pressure levels could still be selected from while facilitating hard press interactions.

Facilitating Other Basic Interactions

According to Buxton's three state model [15] interactions with input devices can be modeled by three basic states: out-of-range (state 0), tracking (state 1) and dragging (state 2). When we consider the state transitions for positioning, single-click, double-click, dragging and clutching, we observe that, with the exception of dragging, all the operations start at state 1 and return to state 1 (Figure 5.10). These can all be handled with the current selection mechanisms that were tested in experiment one. However, dragging necessitates remaining in state 2 and returning to state 1 only when the drag operation is completed.

To model dragging with pressure sensors, pressure buttons would need a mechanism for maintaining the device in state 2. Although many designs are possible and would need to be investigated, we propose two alternatives: *Pressure click-and-hold* and *PressureLock*. With pressure click-and-hold the user would apply pressure, remaining below the hard press activation level. This would result in a switch to state 2 which could be maintained as long as the user maintains pressure. However, fine control over pressure levels can be challenging and PressureLock might be a simpler alternative. PressureLock would allow users to drag and drop items without having to keep the pressure sensor held down while moving the mouse. Once turned on, the user must dwell on the pressure sensor for a brief period when selecting an item to move. Afterwards, the user can release the pressure sensor and drag the item. By tapping on the sensor, the item is dropped at its destination.



Position	1
Single-Click	1-2-1
Double-Click	1-2-1-2-1
Drag	1-2
Clutch	1-0-1

Figure 5.10: State transitions for common mouse operations with a pressure-button mouse described using Buxton's three state model [15].

5.6.3 Design Recommendations

Designers of pressure-augmented mice can take several lessons from our experiments**:**

- The footprint of the pressure buttons must be equivalent to that of mouse buttons in order to improve user accuracy.
- Pressure values in the low end of the pressure space (<300 of 1000 pressure units) are adequate for pressure-based single-clicks and double-clicks.
- The hard press double-click technique is faster than a standard mouse-button double-click.
- Pressure-augmented buttons provide haptic feedback for single-clicks and facilitate hard press double-click actions.

Having completed five studies of two pressure-augmented mouse configurations, we have now provided answers to many of the basic ergonomics and performance questions surrounding the development of pressure-augmented mice. Our research has laid a foundation for future pressure-augmented mouse research, and provided a number of important lessons for commercial designers of pressure-augmented mice. With our primary task complete, we return our attention to the larger problem of providing more expressive interactions to desktop users. In the following chapter, we design and implement six augmented WIMP interactions that leverage the capabilities of a pressure-augmented mouse. We then test these pressure-augmented interactions in a subjective study to determine if a pressure-augmented mouse can improve desktop interactions.

Chapter 6

Application: Augmenting Desktop Interactions

Our research leading up to this point has been concerned with the empirical performance of pressure-augmented mice in low-level experimental tasks. In this chapter we focus on the use of a pressure-augmented mouse in realistic interactions in order to test the effectiveness of our augmented mouse in providing powerful, more expressive interactions.

Using the framework discussed in Chapter Three we designed six augmented interactions for a pressure-augmented mouse. These interactions were tested in a subjective user study using a mouse with the pressure-augmented button design from Chapter Five. This design was chosen because the physical experience of using this pressure-augmented mouse is as close as possible to using a standard mouse; there are no additional buttons on the mouse, the mouse buttons click normally, and expressive pressure input is provided. This device choice was important for our study, as we are primarily interested in testing interaction preferences rather than device preferences.

A design goal for the interactions was to show that additional functionality can be added without changing the traditional behaviour of the interaction: all six of the interactions presented retain their standard functionality, and behave as expected when activated with a standard mouse.

6.1.1 Augmented Scroll Buttons

Scroll buttons are control widgets available on most windows and documents that are too large for their display space. Each scroll button controls scrolling for a single direction on a single axis of the document. Scroll buttons are usually activated by a button click, with a single click scrolling a small distance (a 1D-D action, as defined in section 3.1.2) and a click-and-hold action resulting in a continuous scroll at a fixed rate (1D-D + 1D-D (dwell time)). Time may also be used as a continuous dimension for a scrolling acceleration function. Other similar buttons include other view controls ("zoom in," "zoom out") and application-specific changes to data ("increase contrast," "reduce brightness"). Similar controls have been developed in the past for specific devices [9, 56].

Table 6.1: Augmented scroll buttons as described in terms of the components of the augmented interactions framework in Chapter Three.

	Standard	**Augmentation**
Task	Short Range Scrolling	Mid Range Scrolling
Object	Scroll Buttons	
Action	1D-D (*activation*) 1D-D (*dwell time*)	1D-C (*scroll rate*)
Input	1D-D (*button click*) 1D-C (*time*)	1D-C (*pressure*)

The problem with the continuous mode of a scroll button is that the rate is out of the user's control. Either the rate is fixed and may be too slow or too fast for the user's needs, or is based on time rather than user input. As a result, scroll buttons are only suitable for scrolling short distances in a document, and if the user wants to scroll a larger distance, they must move their cursor to a different widget (the scroll thumb), which may break their concentration on the task.

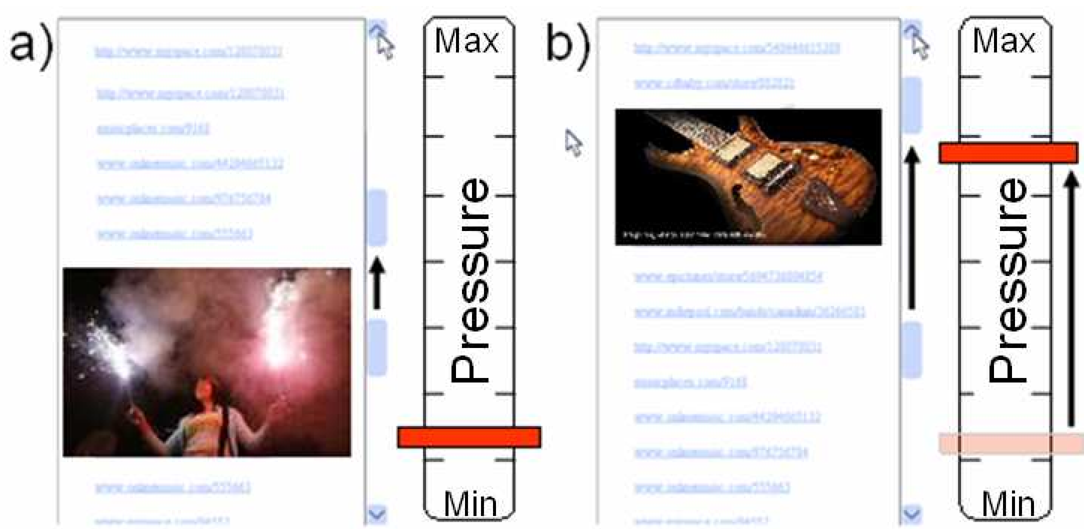


Figure 6.1: a) An augmented scroll button in use. b) As pressure input increases, the document scroll rate increases.

Augmented scroll buttons (Figure 6.1) behave like standard scroll buttons, but in continuous mode also allow the user to choose a continuous scroll rate that is most appropriate for their current task. Augmented scroll buttons can be particularly useful for stylus and touch screen environments where no scroll-specific devices exist. Pressure appears to be a good choice for an augmented scroll button, as pressure maps well to rate control [28, 35].

6.1.2 Augmented Scroll Thumb

The scroll thumb is a control widget available in most applications along with scroll buttons. A scroll thumb corresponds to either the horizontal or vertical dimension of the document and is controlled using a 1D-C DoF from a device, usually one of the axes of the system's pointing device. The high speed of long-distance scrolling, however, presents a problem: the document moves through the viewport too quickly for the user to maintain adequate awareness of their location [31].

Table 6.2: Augmented scroll thumb as described in terms of the components of the augmented interactions framework in Chapter Three.

	Standard	**Augmentation**
Task	Scrolling	Zooming
Object	Scroll Thumb	
Action	1D-D (*activation*) 1D-C (*scroll position*)	1D-C (*zoom level*)
Input	1D-D (*button click*) 1D-C (*mouse axis*)	1D-C (*pressure*)

One solution to this problem was addressed with Speed-Dependent Automatic Zooming [31]. However, this solution couples zoom level to scroll speed, meaning that users do not have direct control over both actions. As first shown in the OrthoZoom technique [4], our augmented scroll thumb allows independent control of both actions from a single widget, allowing more powerful interaction.

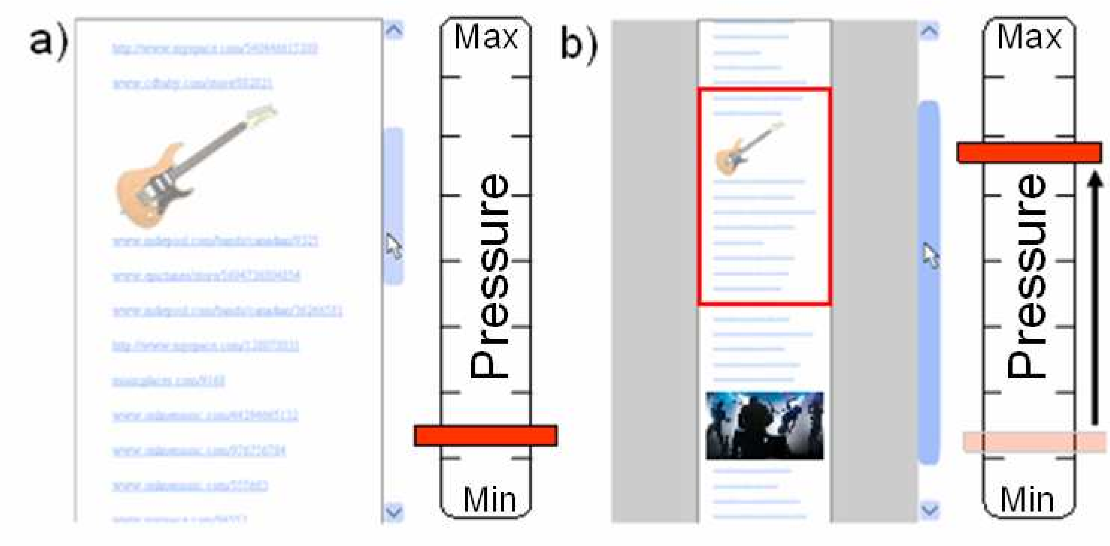


Figure 6.2: a) An augmented scroll thumb in use. b) As pressure input increases, the document zooms out, and the scroll thumb expands accordingly.

OrthoZoom's use of the horizontal mouse dimension for zooming is advantageous in that no extra device or augmentation is required, but using a pressure-augmented button has some advantages. Using the orthogonal dimension may mean that unintended zooming or scrolling can occur due to non-linear mouse movement. Using pressure may make simultaneous scrolling and zooming more difficult [35], but should ensure that unintended scrolling or zooming does not occur. However, pressure input is not bidirectional, so zooming in cannot be controlled with the augmented scroll thumb.

6.1.3 Augmented Object Previews

Web links and icons for files and file folders are object widgets that represent underlying data, but do not fully describe it. For example, a file icon shows the type and name of a file, but not its contents; hyperlinks on web pages show even less, often indicating only that a link exists. Objects may include a number of discrete preview states, but the lack of complete information may mean that users must open the file, or traverse the hyperlink, in order to determine whether the object was the correct one for their purposes. In many situations, this detailed inspection of the object takes considerable time, since the file or link must be fully loaded before their contents can be inspected.

Table 6.3: Augmented object previews as described in terms of the components of the augmented interactions framework in Chapter Three.

	Standard	**Augmentation**
Task	Previewing	
Object	Web Links	
Action	1D-D (*selection*)	1D-C (*preview size*)
Input	1D-D (*button click*)	1D-C (*pressure*)

With augmented object previews (Figure 6.3) the amount of preview information given is controlled by the user. This allows the user to select how much preview information is appropriate, avoiding the problem of presenting too much or too little preview information to the user and reducing demands on system resources such as bandwidth and display real estate. The type of preview information should be specific to the type of object. Preview information can be given through a scalable thumbnail, a status bar with several discrete levels of specific information, or audio or video clips.

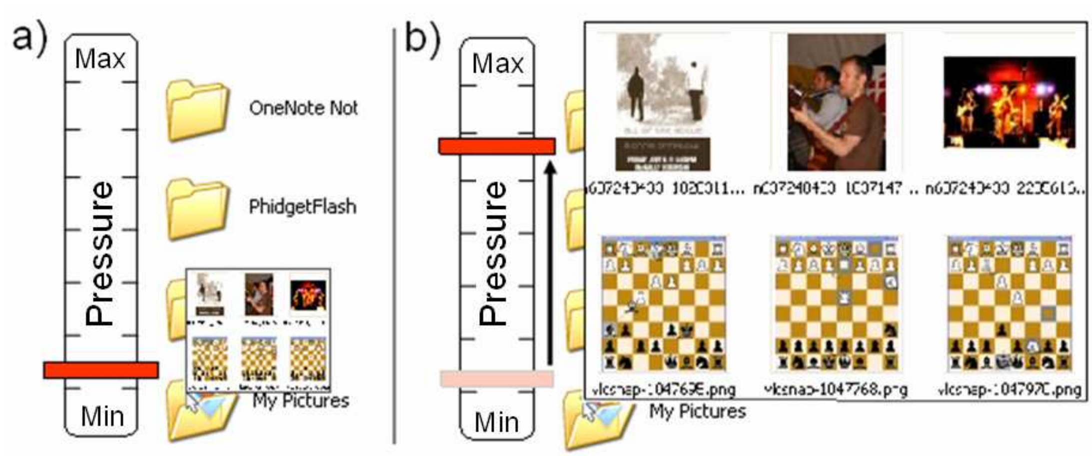


Figure 6.3: a) An augmented preview of a file folder. b) As pressure input increases, the preview thumbnail size increases.

A pressure-augmented mouse button is theoretically not the best choice for this type of interaction; pressure can be difficult to maintain [49, 55, 57], making previewing difficult over extended periods. However, pressure may map well to an abstract parameter like user interest, making for a more natural mapping than an initial analysis suggests.

6.1.4 Augmented Back/Forward Buttons

Back and forward buttons are navigational controls commonly presented in web browsers and file explorers. When activated with a click they typically invoke a single action, and are usually accompanied by a pull-down history menu or history window to facilitate jumps of several pages at a time. History menus can assist user recognition before traversing to a certain page, but occupy additional screen real estate and require the user to switch their attention to (and acquire with the mouse) another widget, breaking concentration on the task. When focused on a task, users may carry out several back button activations rather than switch to the menu-based control.

Table 6.4: Augmented back/forward buttons as described in terms of the components of the augmented interactions framework in Chapter Three.

	Standard	**Augmentation**
Task	Traverse 1 Page	Traverse 1-3 Pages
Object	Back/Forward Buttons	
Action	1D-D (*activation*)	1D-D (*pages traversed*)
Input	1D-D (*button click*)	1D-C (*pressure*)

With augmented back and forward buttons (Figure 6.4), the user can traverse one or many pages with a single control object. This merges the back and forward buttons with a history menu of the most recent traversals.

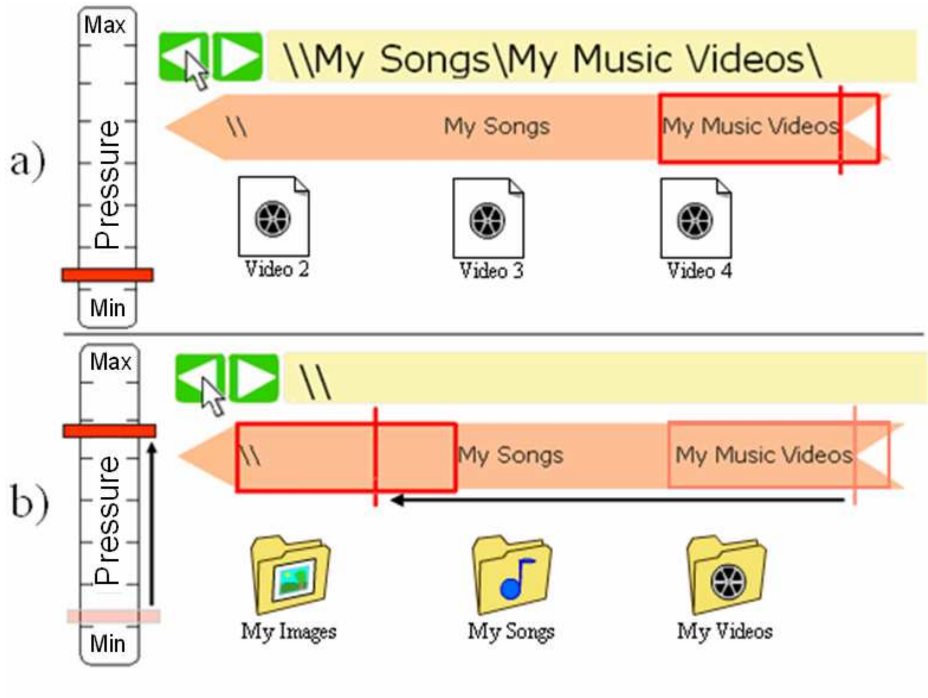


Figure 6.4: a) An augmented back button. b) As pressure input increases, additional pages are traversed. Visual feedback shows the user what pages the user can traverse back to.

In this implementation continuous pressure input is discretized: our own research in Chapter Four, as with other research, indicates that augmented back and forward buttons should traverse a maximum of six pages [49, 55].

6.1.5 Augmented Activations

Objects like file icons and web links have default actions associated with them, representing an activation of the object or an opening of the object's representative data. However, some objects represent sensitive or possibly dangerous content, such as system-critical folders, or web pages and file downloads that have been identified as potentially harmful. User preferences and system security settings often require that users confirm such activations through a confirmation dialog box, or may even require that users activate several menus to alter their preferences or security settings. This can result in user frustration, especially when a user's task is interrupted and when objects that the user knows are safe have been marked as potentially harmful.

Table 6.5: Augmented activations as described in terms of the components of the augmented interactions framework in Chapter Three.

	Standard	**Augmentation**
Task	Opening a Folder	
Object	File Folders	
Action	1D-D (*selection*)	1D-D (*confirmation*)
Input	1D-D (*button click*)	1D-C (*pressure*)

With augmented activations (Figure 6.5), a user is able to specify a confirmation action as they activate an object, making a one-time change to their preferences or security settings and allowing the user's task to continue without interruption. Although double-clicking is typically used to activate an object, this augmentation was applied to a single click. To open the folder, the user applies pressure beyond a fixed threshold. Clicking or double-clicking alone does not open the folder, and instead feedback such as a confirmation dialog is given.

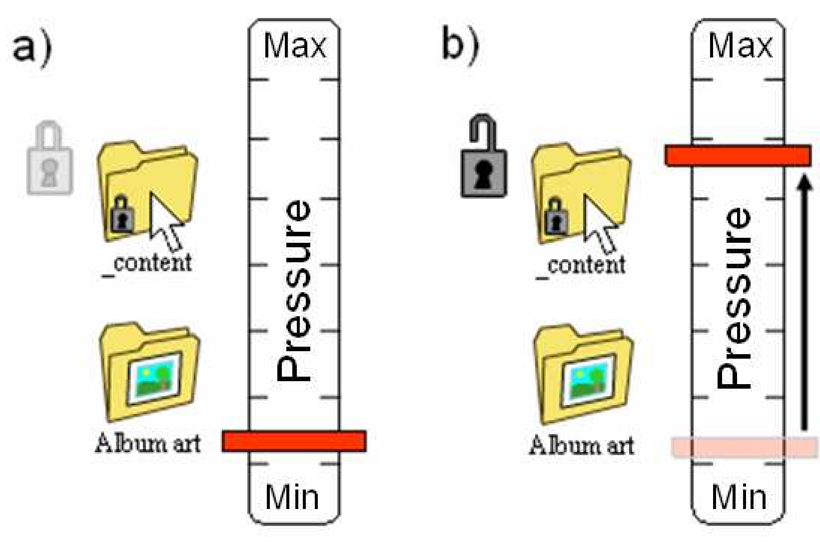


Figure 6.5: a) A file folder that can be activated with an augmented activation. b) When pressure input crosses a threshold the folder is opened.

Pressure input can be discretized to provide multilevel selections similar to a pop-through mouse button [21, 70]. As well, the extra effort required to apply additional pressure may provide a natural mapping to user confidence and interest, and also serve as a physical barrier to prevent accidental augmented activations.

6.1.6 Augmented Dragging

Dragging is an action (1D-D + 2D-C) that is performed on objects and controls within a GUI. A drag action is usually initiated by selecting an object or group of objects by depressing a primary button on a pointing device, performing a pointing action and then releasing the button to end the drag. With the primary button depressed the user is unable to perform other tasks that require the primary button, forcing the user to make any confirmation actions after the drag is completed, or to perform multiple separate drag actions and folder traversals when moving a file several levels through a file system.

Table 6.6: Augmented dragging as described in terms of the components of the augmented interactions framework in Chapter Three.

	Standard	**Augmentation**
Task	Moving a File	
Object	File Icons	
Action	1D-D (*selection*) 2D-C (*drag*)	1D-D (*confirmation*)
Input	1D-D (*button click*) 2D-C (*mouse axes*)	1D-C (*pressure*)

Augmented dragging (Figure 6.6) allows the user to perform other actions while in the middle of a drag operation. In our implementation a user can confirm a file move while dragging onto a folder, but other applications include traversing folders without dropping the files they are dragging (as with ‘spring-loaded’ folders), or dragging a window into view before dropping the dragged files.

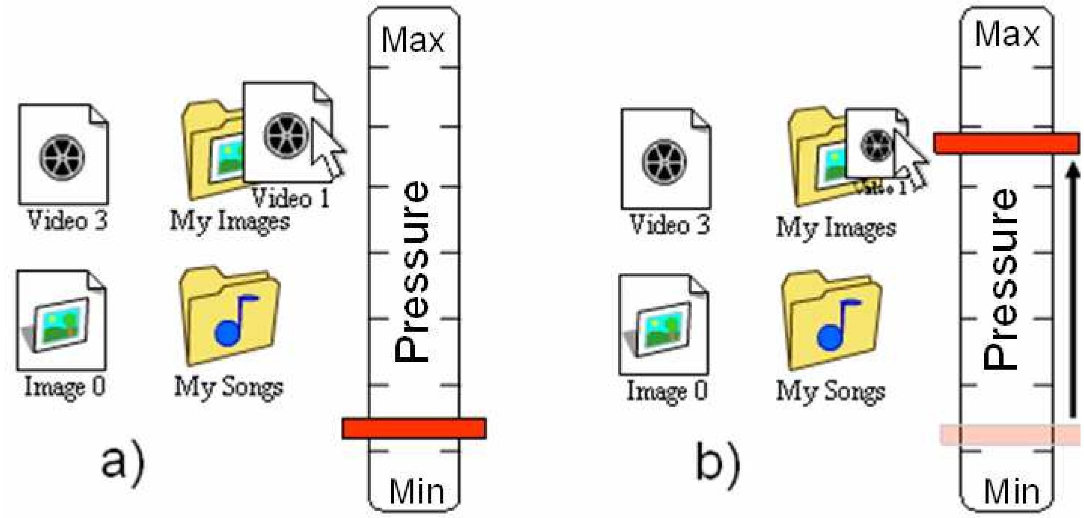


Figure 6.6: a) The user has configured a folder to only accept image files. b) When pressure input crosses a threshold, the file shrinks to indicate it can be moved, overriding the default.

Similar to pop-through buttons [70], pressure input can be discretized to allow multilevel actions. As with augmented activations, the extra effort required to apply additional pressure may serve as a physical barrier to prevent accidental actions while dragging.

6.2 User Study of Augmented Interactions

To determine if augmented interactions can improve desktop interfaces, we performed a subjective user study of the six augmented interactions described above. Our primary goal was to determine if the augmented interactions would be seen as valuable. We were also interested in how the pressure-augmented mouse would affect people's experience with the designs.

6.2.1 Study Apparatus

The user study was conducted using a simulated web browser and a simulated file explorer, both built in Flash. Each application implemented three of the six augmented interactions. The pressure-augmented button was implemented as described in section 5.6.2, by inserting a pressure sensor (CUI Inc., IESF-R-5L) under the primary button of a Microsoft Comfort Mouse. This configuration did not interfere with normal clicking operations, and reported pressure values after the primary button was depressed. The pressure sensor reported a digital value from 1 to 1000 through a Phidgets [22] interface. The test applications were run on a P4 3.2 GHz PC running Windows XP with an LCD monitor at 1024x768 pixels.

The web browser application included the augmented scroll buttons, the augmented scroll thumb, and the augmented object previews, as described above. The file explorer application implemented the augmented back/forward buttons, the augmented activation mechanism, and augmented dragging (described above).

The scroll thumb behaved normally when activated with a button click. The zoom level of the document was controlled by the pressure applied to the mouse button while the scroll thumb was activated. Pressure was mapped continuously to zoom level such that the maximum pressure level of the sensor zoomed the document to 50% size.

The web browser links displayed a 15% thumbnail preview when activated with a button click. The size of the thumbnail was controlled by the pressure applied to the mouse button while over the link. Pressure was mapped continuously to thumbnail size such that the lowest level of pressure scaled the thumbnail to 15% size and the highest level of pressure scaled the thumbnail to 50% size.

The file explorer test included augmented back/forward buttons, augmented activation and augmented dragging. The back and forward buttons behaved normally when clicked. When extra pressure was applied, an arrow-shaped widget (Figure 6.4) appeared displaying the current pressure level with a sliding pressure cursor [55], and the recent pages that were available to be visited. Additional pressure allowed selection of a page up to three back in the list.

Secure folders were identified using a lock icon. When clicked, a partly-transparent lock would appear next to the folder indicating to the user that the folder was locked. By applying additional pressure to the folder the lock became less transparent until a threshold was crossed and the lock was opened. The folder was opened with the button release.

Folders were set to only accept files of a specific type, identified by their matching icons. A file dropped on a folder of a differing type would slide off and not be moved into the folder. If pressure was applied past a threshold, the file icon would shrink to 50% of its size, indicating to the user that the file would be moved into the folder if released.

6.2.2 Participants and Tasks

Ten participants (seven men, three women) between the ages of 19 and 34 were recruited. Five participants were computer science students, four were students from other disciplines, and one participant was a full-time employee. All participants were experienced with WIMP GUIs, and seven of the participants had experience using pressure-sensing devices. All participants used the mouse in their right hand.

Participants were shown the two demonstration applications, were given a brief description of each augmentation, and were introduced to the pressure-sensitive button on the mouse. Participants were asked to explore the application and functionality of the interactions and were occasionally prompted with tasks. Participants were encouraged to voice opinions during the study and notes were taken by the experimenter. After approximately ten minutes with each application participants were asked to rate each of the interactions in terms of five qualities: ease of learning, ease of use, efficiency compared to the standard, suitability of pressure to the interaction, and desirability of the new functionality. In the case of augmented activation and augmented dragging, subjects were asked to compare against a confirmation dialog. Ratings were given on a five-point scale. Participants were also asked to state their preferences.

6.2.3 Results

Here we discuss the results of our evaluation of augmented interactions. For a complete table of results see Figure A.7.

Overall reaction to augmented interactions was positive, with 9 of 10 participants saying the new functionality of augmented interactions was desirable (4 on a 5 point scale). Interaction by interaction the desirability ratings were high, with 41 of 60 ratings being a 4 or higher, and only three ratings in total that were a 1 (very undesirable) or 2 (undesirable).

The most preferred interactions were augmented activation, augmented dragging, and augmented scroll bars, with 7 of 10 participants preferring them to the standard. The least preferred were augmented back/forward buttons with five participants preferring them to four participants preferring the standard (one having no preference) and augmented scroll buttons, with four participants preferring them to three preferring the standard (three having no preference).

From participant comments during the study and in the questionnaire there seem to be two contributing factors for the low preference ratings that augmented scroll buttons and augmented back/forward buttons received. The first has to do with our specific implementations. Three participants commented that they would like to preview the pages with the augmented back/forward buttons as they applied and released pressure, suggesting they would have preferred an implementation with a selection action. Three participants also commented that the augmented scroll buttons did not scroll as fast as they wanted them to, suggesting that the maximum scroll speed should be increased. The second reason for the preference ratings relates to pressure control in general, which we discuss later in this section.

Participants strongly agreed that augmented interactions were easy to learn. Only one interaction (augmented object previews) received a single neutral ease-of-use rating; all remaining ratings for all interactions were 4 or 5 (‘easy to learn’ and ‘very easy to learn’). Also, the majority of augmented interactions received high ease-of-use ratings. Augmented activations, scroll thumbs, and dragging each received only one rating less than 4, and augmented scroll buttons and augmented previews received only two ratings less than 4.

The most common usability problem reported was difficulty in applying pressure. Five users mentioned that they felt they had to apply too much pressure to reach the maximum activation point of the sensor. This is due in part to the design of our pressure-augmented button: the plastic casing over the mouse button has the effect of damping the pressure sensor underneath, requiring users to press harder than they would if using the sensor alone. A pressure-sensitive mouse button design with the pressure sensor included in the button mechanics would likely improve usability and reduce the overall difficulty of applying and maintaining pressure on the button. In spite of the comments made by users regarding difficulty applying pressure, users still rated augmented interactions highly in terms of ease of use, suggesting that pressure control is not a significant drawback to the interactions.

6.3 Discussion

The user study indicates that users can learn the augmented interactions quickly, and that they find the interactions simple and easy to use. In addition, almost all of our participants were strongly in favour of augmented interactions, stating they would like to use them in their everyday work. These results demonstrate that pressure-augmented mice can be used to improve desktop interactions. In the sections below, we discuss the results of the study in terms of the design framework, the relationship of our framework to other models, and issues regarding the deployment of augmented GUI interactions.

6.3.1 Prediction of Results Based on the Framework

Though subjective results for the interactions were generally positive, the neutral and negative results tend to correlate with how appropriate pressure is for controlling the action. For instance, augmented back/forward buttons scored relatively low in ease of use and preference; a number of papers discuss how force sensing devices perform poorly for precision selection tasks [49, 55, 57], and our results in Chapter Four also show a large number of errors. However, with some modifications, this type of pressure-based selection may perform better [62].

The results show that augmented activation, augmented dragging, and augmented scroll thumb were most preferred. Augmented activation and augmented dragging were predicted to perform well since they are multi-layer interactions which a pressure-augmented button well supports. These two interactions may have also performed well due to the natural mapping of pressure to ‘increased intention or confidence’. It is also likely that users perceive the benefit to avoiding confirmation dialogs when possible.

6.3.2 Deployment and Adoption of Augmented Interactions

Our experiences with augmented interactions suggest that more augmented interactions could be successful in standard applications. Several questions arise, however, when considering wider-scale deployment:

Will input hardware support the new designs?

Pressure-augmented mice, if they are made commercially available, could be used to control augmented interactions. However other input devices could also be used. Additional degrees of freedom are constantly being added to input devices: scroll wheels are now standard, commercial devices such as the IBM ScrollPoint mouse and the Xbox 360 controller support pressure input, and devices like isometric joysticks, pressure sensors, and multitouch screens are widely available. As powerful input devices become readily available, more applications in a variety of contexts can make use of their capabilities using our framework.

Will new designs break existing interaction styles?

A main design goal for our augmented interactions was to improve task support without removing the original interaction. Augmented GUI applications will not alienate users that do not want to use the augmented functionality, or do not have access to a required device, because the default behaviour of the new designs has not been changed.

Will users like the new designs in real-world use?

The strong positive response in the user study suggests that augmented interactions have considerable promise. In particular, techniques such as the ‘press hard to confirm’ augmented activations were seen as both novel and valuable by participants (showing that the framework can actually aid GUI innovation). In addition, many of the participants were able to suggest other objects and controls that could be improved through augmentation, indicating that the basic idea of augmented GUIs has wide applicability.

6.4 Summary

The results of our subjective user study show that a pressure-augmented mouse can be used to control more powerful and expressive interactions. Results also demonstrate that augmented interactions with the pressure-augmented mouse were highly desirable, easy to learn, and easy to use. This study demonstrates that pressure-augmented mice produced using information from our research can improve the desktop computing experience. With our primary and secondary research goals complete, we now present a discussion of our results and findings, followed by a summary of this thesis.

CHAPTER 7

DISCUSSION

Here we elaborate on the findings of our studies examining pressure-augmented mice and augmented interactions. We begin by discussing the implications of our findings for both mouse designers and interface designers, and then discuss the most important design lessons that were learned through the course of our research.

7.1 Summary of Findings

Results from five quantitative user studies and one subjective user study have yielded a number of important results. Here we summarize our most important findings.

7.1.1 Mouse-Based Pressure Control is Similar to Stylus-Based Pressure Control

Our uni-pressure mouse study in Chapter Four was modeled after a similar study performed with a pressure-sensing stylus [55]. The results of Ramos and colleagues' [55] research agree with the results of our uni-pressure mouse study, suggesting that users can control up to six levels of pressure when given full visual feedback. Both research studies found that users were fastest when the number of pressure levels was six or less [55]. However, our results reveal some important differences between pressure-sensing mouse input and pressure-sensing pen input.

In their research with pressure-sensing pens, Ramos and colleagues found no significant difference in the number of crossings between four and six pressure levels [55] while our uni-pressure mouse research found a significant difference in crossings between four and six levels. However, our study demonstrated that there was no significant difference in movement times between four and six pressure levels. This combination of results may suggest that users have more precise pressure control with a stylus. However, in spite of an increase in crossings when the number of pressure levels increases from four to six, user selection speed with a pressure-sensing mouse was not affected. Our results also differ with the results presented by Ramos and colleagues [55] in terms of the most effective selection mechanism. While their results show that the QuickRelease mechanism is fastest [55], our results show that the *click* mechanism is fastest (Figure 4.6). This difference in results can be accounted for by the differences in the form factor and use of the devices. For instance, the QuickRelease mechanism with a pen requires a movement of the wrist or arm away from the tablet, which is a more natural movement than quickly lifting a single finger from a gripped object like a mouse. Similarly, the Click mechanism with a stylus requires that a user press the side button on the stylus while maintaining constant pressure. This action is difficult since the finger that is used to press the button is also used for maintaining a grip on the stylus, and clicking the side button may change the amount of pressure applied downwards through the pen. Using the *click* mechanism with a mouse requires maintaining pressure with one finger and clicking with another, which is comparatively easy. The results of both our study and the study performed by Ramos and colleagues [55] agree that the dwell selection mechanism is the most accurate in terms of the number of errors [55]. (See section 4.3.2).

7.1.2 Dual-Pressure Input Can Be Used to Control 64 Discrete Levels

Because the form factor of a mouse can accommodate multiple sensors, our research in Chapter Four went beyond the research of Ramos and colleagues [55] and tested dual-pressure input. The results of this study show that when the number of pressure levels increases beyond 12, it is more efficient to use dual-pressure input (Figure 4.9). Because this research study was intended to highlight the movement time cross-over at 12 pressure levels, we tested only one large discrete configuration of 64 levels (8x8 pressure levels). Other configurations that would be useful to test include 36 discrete levels (6x6 pressure levels) and 48 discrete levels (6x8 and 8x6 pressure levels). Though we expect that a smaller number of discrete levels would produce faster selections times, our results indicate that the controllability of dual-pressure input is consistent for a discrete number of levels between 12 and 64; we found no significant difference in the number of crossings with the *tap-and-refine* technique for 12, 16, and 64 discrete levels (Figure 4.10).

One of the applications that Ramos and colleagues [55] mention for stylus-based pressure widgets is pressure-controlled menus [55, 73], but many menus include more than six elements and some even contain large tree structures. With dual-pressure input, and 64 discrete levels being controllable, a far greater number of elements could be included in Pressure Widget menus [55]. Additionally, a slightly modified *tap-and-refine* technique could be used to navigate tree structures.

7.1.3 Pressure Input Can Efficiently Replace Double-Clicks, But Has Error Issues

The results of our second study of pressure buttons in Chapter Five showed that users are significantly faster when using a single hard press action than when using a standard mouse button double-click for action invocations. Similar results were also found in our study of pressure-augmented buttons. Using a pressure button or pressure-augmented mouse button in this way is similar to using a pop-through button [70] in that the pressure button effectively has two depressed states. Pressure-sensor buttons also have the advantage of providing continuous parameter control and can also facilitate discrete selections from more than two states.

However, like our other pressure-based selection and action mechanisms, our hard press double-click technique suffered from a large number of errors. As discussed later in this chapter, error rates must be reduced before hard press interactions could be used in practice.

7.1.4 Users Prefer More Powerful and Controllable WIMP Interactions

The subjective results of our augmented interactions study revealed that users want more powerful and controllable GUI objects and controls like the six example augmented elements we tested. Nine out of ten participants from our study answered that the functionality of the new GUI widgets was desirable or very desirable, with the remaining participant answering neutrally. In terms of user preferences regarding each of the six elements, more users preferred the augmented versions than the standard versions of the elements. Users agreed that the functionality of the new augmented elements was easy to learn and to use; except for a single neutral rating of one of the elements, all the ease of use ratings for all the augmented elements were a 4 out of 5 or higher (‘easy to learn’ or ‘very easy to learn’). Usability ratings were also generally high, with only 6 ratings out 60 being less than a 4 (‘easy to use’).

7.2 Lessons Learned

Although the results of our experiments with pressure-augmented mice have produced a number of lessons for designers, several of our findings were particularly surprising or noteworthy. Here we summarize the most important lessons we learned from our experiments.

7.2.1 Pressure Input is Relatively Imprecise

Although pressure input can be used to control discrete selections, the number of crossings and number of errors recorded in our experiments indicate that pressure input is relatively difficult to precisely control. In our first study of Chapter Four we found that users made an average of 1.3 crossings per trial. In that same study we found that the average error rate for the *click* technique was 0.11 errors per trial for 4 pressure levels and 0.19 errors per trial for 6 pressure levels. In the second study of Chapter Four we found that the *tap-and-refine* technique had an average error rate of 0.17 errors per trial. In our first and second studies of Chapter Five we found that pressure-based single and double-click techniques both had error rates of 0.06 errors per trial. Even for small numbers of discrete pressure levels, error rate may prohibit the use of pressure as a discrete selector, especially in interactions where the penalty for errors is high.

However, our results with the *dwell* technique (0.01 errors per trial) indicate that there is hope in improving the controllability of pressure input. The extremely low error rate of the *dwell* technique demonstrates that users are able to maintain pressure on a specific pressure level for at least 1 second, though at the expense of selection time and number of crossings. With further research, it should be possible to reduce the number of crossings and errors for all techniques and make pressure input a more precise and controllable form of input. Some promising research has already been performed [62], demonstrating that more sophisticated pressure distribution functions can improve the controllability of pressure.

7.2.2 Effective Pressure-Sensor Buttons Must Have a Large Enough Footprint

The effective sensing area of the pressure sensors used in our experiments is small compared to the clickable area of a standard mouse button. The IESF-R-5L sensor from CUI Inc. has a circular area with a diameter of 10 millimetres, but the sensing area is smaller, with a diameter of 8 millimetres. By observing the participants while they performed the first and second experiments discussed in Chapter Five we noticed that errors were often caused by the user not pressing directly in the central sensing area of the sensor as they made their ‘clicking’ motion. The small footprint area of the sensor may also be partly responsible for the low subjective rankings of the pressure-based clicking and tapping techniques.

To improve the performance of a pressure button, sensors with larger sensing areas must be produced and tested against standard mouse buttons. These pressure sensors would ideally have a much larger area than the IESF-R-5L sensors and would occupy a space similar in size to a standard mouse button. While these sensors would no longer be subject to the problems associated with a small footprint, a larger sensing area may cause other problems. For instance, most users rest their fingers on top of the mouse buttons when using a mouse; a sensor with a very large area would need to be calibrated appropriately to avoid sensing unintended clicks from fingers resting on them. Such calibrations may be difficult, as research indicates that users apply additional pressure to mice in stressful situations [54].

Pressure buttons are an achievable technology, and working around the current issues of footprint and lack of tactile feedback are possible to a certain extent. However, it should also be noted that the most obvious solution to both problems is to implement pressure-augmented buttons, which have no footprint issues, supply tactile button-clicking feedback, and should provide a level of pressure control similar to the results of our studies in Chapter Four if appropriately designed.

7.2.3 Pressure Maps Well to User Interest and User Confidence

One of the most interesting interactions that we explored in our design and implementation of augmented interactions in Chapter Six was the use of pressure as a measure of abstract parameters like user interest and confidence.

With augmented object previews (section 6.1.3), pressure is mapped to user interest. To view more information about objects like folders, files, and hyperlinks, the user increases pressure input. This type of interaction allows objects to be designed with a minimal amount of default information displayed. Though we only explored this interaction in terms of typical desktop style interactions, this mapping of pressure to user interest could also be applied to distributed groupware applications. For instance, Birnholtz and colleagues [11] developed a groupware chat application that displays a variable amount of awareness information for each user. To gain more information about a particular user's current state and activities a user can 'pull' open their chat representation to reveal more information. The intention of this interaction is to allow for increased awareness for a limited time at the price of extra effort from the user, thus making the interaction perfect for pressure input to control. Pressure input requires some physical exertion from the user, and the reflexivity property of pressure sensors ensures that users must release the interaction after a finite time.

Augmented dragging (section 6.1.6) and augmented activations (section 6.1.5) both map pressure to user confidence. When performing an action that would normally be blocked or require confirmation (e.g., through a confirmation dialog box), the user is allowed to perform the desired action immediately by increasing pressure input. Both augmented dragging and augmented activations allow users to override system settings or user preferences by increasing pressure input when they perform an action. This interaction style has two main advantages: Firstly, it allows users to bypass their preferences or security settings on-the-fly without having to alter them permanently. Secondly, it allows users to confirm an action as they make it, avoiding unnecessary confirmation dialog boxes which can break up and disrupt the user's task flow. This particular interaction style could be particularly useful in speeding up tasks once a user is familiar with a system and knows in advance when they would receive a confirmation dialog. Pressure input is particularly effective for this interaction technique because it requires minimal effort to perform, but the extra physical effort required helps ensure that the interaction is not performed accidentally. Pressure provides a natural mapping for confidence and intent, and could be used in a number of interfaces where a measure of user confidence is valuable [21].

7.2.4 Users Want More Powerful GUI Interactions Regardless of Input Device

It is important to note that while we designed our six example augmented interactions for a pressure-augmented mouse button, augmented interactions could be designed to work with nearly any input device. One particular experience from our study of augmented interactions in Chapter Six demonstrates that the power of augmented interactions is in their added control and functionality, and not necessarily pressure input. While performing the study, one participant expressed great frustration due to the amount of pressure required for maximum activation of the sensor. After completing their exit questionnaire the participant was asked to try the web browser test application again using a Phidgets slider [22] instead of the pressure-augmented mouse button. After trying the three web browser augmented interactions with the slider the participant's reaction was much more positive, stating that they preferred the augmented scroll buttons and augmented object previews with a slider to both the standard versions and the augmented versions with pressure. Since augmented interactions are not limited to pressure interactions only, this example demonstrates that an appropriate input device must be chosen based on the capabilities of the user; users may have difficulty applying pressure, performing bimanual interactions, or performing other physical tasks, limiting their ability to use certain input mechanisms. Unlike other device-specific interaction techniques, an augmented interaction can be designed to work with a wide range of input devices. Given this particular participant's positive reaction to the augmented widgets when using a more preferred device, and the overall ratings of augmented interactions in terms of desirability and user preference, our results strongly indicate that users want more power in their GUI interactions, regardless of the specific input device used.

CHAPTER 8

CONCLUSION

The problem examined in this thesis was: there is a lack of ergonomics and performance information to aid the design of pressure-augmented mice. Our solution was to provide empirical performance and ergonomics information as well as design recommendations for pressure-augmented mice. We performed five experiments to test design parameters for two different pressure-augmented mouse configurations. By comparing empirical results from these experiments, we were able to identify the most effective parameters for each mouse configuration and provide design recommendations for future pressure-augmented mice. In a follow-up evaluation we demonstrated that augmented interactions controlled by a pressure-augmented mouse are usable and desirable.

8.1 Contributions

8.1.1 Primary Contributions

There are two primary contributions presented in this thesis: the results of our experiments in Chapters Four and Five that provide performance information for the design of pressure-augmented mice, and the design recommendations for pressure-augmented mice and pressure buttons that were developed from the results of our experiments. Here we present our design recommendations once more.

Design Recommendations for Pressure-Augmented Mice

As discussed in section 4.7, there are five main lessons that mouse designers can take from the results of our experiments with pressure-augmented mice.

- Place pressure buttons so that they are accessible by the middle finger and the thumb.
- Use mouse button clicks for confirming pressure-based discrete target selections.
- Limit the number of pressure levels selectable with a single pressure sensor to six.

- Use dual-pressure mechanisms for increasing the selectable range of pressure levels and modes.
- *Tap-and-refine* is the most effective control mechanism for providing bidirectional pressure input and controlling a large number of pressure levels.

Design Recommendations for Pressure-Sensor Buttons

As discussed in section 5.6.3, there are four lessons that mouse and pressure button designers can take from the results of our experiments with pressure-sensing buttons.

- The footprint of the pressure buttons must be equivalent to that of mouse buttons in order to improve user accuracy.
- Pressure values in the low end of the pressure space (<300 of 1000 pressure units) are adequate for pressure-based single-clicks and double-clicks.
- The hard press double-click technique is faster than a standard mouse-button double-click.
- Pressure-augmented buttons provide haptic feedback for single-clicks and facilitate hard press double-click actions.

8.1.2 Secondary Contributions

The Design Framework for Augmented Interactions

The framework we present in Chapter Three for augmented interactions can be used to aid the design of more powerful WIMP interactions. As demonstrated, our framework can produce novel interactions and previously studied interactions, and the framework can also be used to compare and contrast different augmented element designs. Our framework is a tool for developing interactions in a systematic way, and while we present the framework in the context of WIMP interactions, it can be utilized in a number of design paradigms [6, 7, 36, 63].

The Six Augmented Interaction Designs

The six augmented interactions that were designed and tested in Chapter Six were inspired by similar interactions for other devices [4, 21, 55]. However, each interaction was designed with the capabilities of a pressure-augmented mouse in mind. None of these interactions had previously been designed or tested with a pressure-augmented mouse. These augmented interactions are described in sections 6.1.1 through 6.1.6.

The Seven Factors that Effect Pressure-Augmented Mouse Performance

The factors that can affect performance with a pressure-augmented mouse in a discrete target selection task (number of levels, sensor location, number of sensors, discretization of raw pressure values, pressure control mechanism, selection mechanism, and visual feedback) had not been identified before (see sections 4.1.1 through 4.1.7). While we used this list of factors to identify parameters for our experiments, they can also be used to identify areas where future research can improve pressure-augmented mouse interactions (e.g., discretization function and pressure control mechanism). Our own research in Chapter Four improved the number of controllable pressure levels through dual-pressure input.

Two Dual-Pressure Control Mechanisms

In our second study of Chapter Four we designed and implemented two dual-pressure control mechanisms to increase the number of discrete levels selectable with pressure input. These designs are discussed in section 4.4.1 and 4.4.2. To our knowledge, dual-pressure mechanisms for discrete target selections have not be developed or studied before.

8.2 Future Work

Our research has laid the foundation for future pressure-augmented mouse research, and opened a number of avenues for future research with pressure input.

8.2.1 Discretization and Linearization Functions

As identified by CUI Inc., the model IESF-R-5L sensor that was used in our studies has a non-linear response. Similarly, other researchers comment that stylus-based pressure is non-linear in response [49, 55, 56, 57]. To compensate, we used a quadratic function centered at the highest pressure level (see section 4.1.4). However, there is likely a more effective function that could be developed and used. Some research in this area has already begun [62]. Also, while our function works well to stabilize pressure control for discrete target selections, we did not test the function in any continuous input control tasks. More research and testing must be done to find the best linearization function for both discrete and continuous pressure input control.

8.2.2 Reducing Errors

Results from our studies with pressure-augmented mice and pressure buttons show a relatively high error rate. In our first study with pressure-augmented mice the *click* technique was the fastest; however, *click* also had at least 0.11 errors per trial. While the *dwell* technique had a much lower error rate (0.01 errors per trial), it was also the slowest technique. This accuracy/speed trade-off has been discussed by other researchers of pressure input [62]. Pressure buttons also experienced a large number of errors, with an average of 0.06 errors per trial for both single and double-clicks. This number is significant when you consider that it is the clicking task alone that has 0.06 errors per trial; the error rate for full mouse targeting tasks would likely be much higher.

Improving accuracy for pressure-augmented mouse and pressure button interactions is a necessary step in making the interactions more usable. Some attempts have already been made to reduce error rates, with limited success; Shi and colleagues developed a technique for improving accuracy with pressure, based on the metaphor of a fisheye lens [62]. While their techniques represent a first step for improving accuracy, they still report error-rates in excess of 0.20 errors per trial when the number of pressure levels is greater than 6. Improving accuracy with pressure remains an important area of future research.

8.2.3 Prototype Design and Testing of Pressure Button and Augmented Button Mice

The quantitative and qualitative results of our experiments with pressure-sensor buttons and pressure-augmented buttons demonstrate the need for better designed pressure button and pressure-augmented mice. While our results show that there is promise in both designs, our limited implementations of these devices caused several problems which limit the generalizability of some of our findings. For instance, the small footprint of the sensor caused users to make unintended errors when clicking and tapping the sensors in our pressure-button experiments. As well, in our augmented elements study, placing the sensor underneath the mouse button casing made high levels of pressure difficult to reach. These problems have had consequences on our studies of pressure buttons and augmented elements in both quantitative and qualitative terms. Testing a higher quality pressure-augmented mouse in experiments similar to those we have performed would help confirm our results and identify what aspects of our results were affected by the limitations of our pressure-augmented mouse implementations.

8.2.4 Quantitative Evaluation of Augmented Interactions

While our evaluation of the six augmented GUI elements in Chapter Six demonstrated the potential benefits of augmented interactions controlled by a pressure-augmented mouse, we have no empirical evidence to show that our augmented interactions actually improve user performance. A quantitative evaluation is the next step in augmented interactions research. Evaluating a set of augmented GUI elements is necessary to identify how greatly user performance is affected by the new designs, and to compare the effectiveness of a pressure-augmented mouse with other input devices. Such a study would also help refine the designs of individual augmented interactions, making them more effective and usable.

8.2.5 Pressure Gestures

All gestural input involves the interpretation of a set of variables over time. While gestural input is often associated with high-DoF devices and input similar to human hand gestures, pressure can be used for gestural input as well. For instance, Ramos and colleagues [57] implementation of Pressure Marks and Wobbrock and colleagues' implementation of EdgeWrite [69] for an isometric joystick [68] are both gestural input techniques using pressure. Future work with pressure gestures would involve the testing of the human capacity to control continuous pressure over time, the implementation of a gesture set for pressure input, and the design and testing of a gesture recognizer for pressure gestures. Interesting questions include the gestural shapes that users can best enter with pressure, and whether or not users require visual feedback to perform the gestures. For example, if the pressure gesture set is implemented using a small number of relative pressure levels, it may be possible to enter pressure gestures without visual feedback [49, 55, 57]. The application domain in which the gestures are to be used and the results of pressure control experiments will likely have some effect on the gesture set that is developed.

REFERENCES

[1] Agarawala, A. and Balakrishnan, R. Keepin' it real: pushing the desktop metaphor with physics, piles and the pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1283-1292. ACM, New York, NY, 2006.

[2] Akamatsu, M., MacKenzie, I. S., & Hasbrouq, T. A comparison of tactile, auditory, and visual feedback in a pointing task using a mouse-type device. *Ergonomics*, 38, pages 816-827. Taylor & Francis, London, UK. 1995.

[3] Albinsson, P. and Zhai, S. High precision touch screen interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* pages 105-112. ACM, New York, NY, 2003.

[4] Appert, C. and Fekete, J. OrthoZoom scroller: 1D multi-scale navigation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* pages 21-30. ACM, New York, NY, 2006.

[5] Balakrishnan, R., Baudel, T., Kurtenbach, G., and Fitzmaurice, G. The Rockin'Mouse: integral 3D manipulation on a plane. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* pages 311-318. ACM, New York, NY, 1997.

[6] Beaudouin-Lafon, M. Instrumental interaction: an interaction model for designing post-WIMP user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 446-453. ACM, New York, NY, 2000.

[7] Beaudouin-Lafon, M., Mackay, W. E., Andersen, P., Janecek, P., Jensen, M., Lassen, M., Lund, K., Mortensen, K., Munck, S., Ravn, K., Ratzer, A., Christensen, S., and Jensen, K. CPN/tools: revisiting the desktop metaphor with post-WIMP interaction techniques. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, pages 11-12. New York, NY, 2001.

[8] Beaudouin-Lafon, M. Designing interaction, not interfaces. In *Proceedings of the Working Conference on Advanced Visual interfaces*, pages 15-22. ACM, New York, NY, 2004.

[9] Benko, H., Wilson, A. D., and Baudisch, P. Precise selection techniques for multi-touch screens. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1263-1272, ACM, New York, NY, 2006.

[10] Bewley, W. L., Roberts, T. L., Schroit, D., and Verplank, W. L. Human factors testing in the design of Xerox's 8010 “Star” office workstation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems,* pages 72-77. ACM, New York, NY, 1983.

[11] Birnholtz, J. P., Gutwin, C., and Hawkey, K. Privacy in the open: how attention mediates awareness and privacy in open-plan offices. In *Proceedings of the 2007 international ACM Conference on Supporting Group Work,* pages 51-60. ACM, New York, NY, 2007.

[12] Blaskó, G. and Feiner, S. Single-handed interaction techniques for multiple pressure-sensitive strips. In *CHI '04 Extended Abstracts on Human Factors in Computing Systems*, pages 1461-1464, ACM, New York, NY, 2004.

[13] Bohan, M. and Chaparro, A. To click or not to click: a comparison of two target-selection methods for HCI. In *CHI 98 Conference Summary on Human Factors in Computing Systems*, pages 219-220. ACM, New York, NY, 1998.

[14] Buxton, W. Lexical and pragmatic considerations of input structures. *SIGGRAPH Comput. Graph.* 17, 1, pages 31-37. ACM, New York, NY, January 1983.

[15] Buxton, W. A Three-State Model of Graphical Input. In D. Diaper et al. (Eds), *Human-Computer Interaction - INTERACT '90*, pages 449-456. Amsterdam: Elsevier Science Publishers B.V., North-Holland, 1990.

[16] Card, S. K., Mackinlay, J. D., and Robertson, G. G. The design space of input devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People*, pages 117-124. ACM, New York, NY, 1990.

[17] Clarkson, E.C., Patel, S.N., Pierce, J.S., and Abowd, G.D., Exploring Continuous Pressure Input for Mobile Phones, GVU Tech. Report, 2006. GIT-GVU-06-20. http://hdl.handle.net/1853/13138.

[18] Dillon, R. F., Edey, J. D., and Tombaugh, J. W. Measuring the true cost of command selection: techniques and results. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems: Empowering People*, pages 19-26. ACM, New York, NY, 1990.

[19] Fagarasanu, M., & Kumar, S. Carpal tunnel syndrome due to keyboarding and mouse tasks: a review. *International Journal of Industrial Ergonomics*, 31, pages 119-136, Elsevier Ltd., Kidlington, Oxford, UK, 2003.

[20] Fix, J.D. *Neuroanatomy*, 3rd ed., page 127. Lippincott Williams & Wilkins, Hagerstown, MD, 2002.

[21] Forlines, C., Shen, C., and Buxton, B. Glimpse: a novel input model for multi-level devices. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, pages 1375-1378. ACM, New York, NY, 2005.

[22] Greenberg, S. and Fitchett, C. Phidgets: easy development of physical interfaces through physical widgets. In *Proceedings of the 14th Annual ACM Symposium on User interface Software and Technology*, pages 209-218. ACM, New York, NY, 2001.

[23] Grossman, T., Hinckley, K., Baudisch, P., Agrawala, M., and Balakrishnan, R. Hover widgets: using the tracking state to extend the capabilities of pen-operated devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 861-870. ACM, New York, NY, 2006.

[24] Guimbretière, F., Martin, A., and Winograd, T. 2005. Benefits of merging command selection and direct manipulation. *ACM Trans. Comput.-Hum. Interact.* 12, 3, pages 460-476. ACM, New York, NY, September 2005.

[25] Guimbretiére, F. and Winograd, T. FlowMenu: combining command, text, and data entry. In *Proceedings of the 13th Annual ACM Symposium on User interface Software and Technology*, pages 213-216. ACM, New York, NY, 2000.

[26] Harada, S., Saponas, T. S., and Landay, J. A. Voicepen: augmenting pen input with simultaneous non-linguisitic vocalization. In *Proceedings of the 9th*

international Conference on Multimodal interfaces, pages 178-185. ACM, New York, NY, 2007.

[27] Hinckley, K., Sinclair, M., Hanson, E., Szeliski, R., and Conway, M. The VideoMouse: a camera-based multi-degree-of-freedom input device. In *Proceedings of the 12th Annual ACM Symposium on User interface Software and Technology*, pages 103-112. ACM, New York, NY, 1999.

[28] Hinckley, K., Input Technologies and Techniques, in *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*, ed. by A. Sears & J. Jacko, pages 3-11. Lawrence Erlbaum Associates, Inc., 2002.

[29] Hinckley, K., Cutrell, E., Bathiche, S., and Muss, T. Quantitative analysis of scrolling techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 65-72. ACM, New York, NY, 2002.

[30] Hourcade, J. P. and Berkel, T. R. Tap or touch?: pen-based selection accuracy for the young and old. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, pages 881-886. ACM, New York, NY, 2006.

[31] Igarashi, T. and Hinckley, K. Speed-dependent automatic zooming for browsing large documents. In *Proceedings of the 13th Annual ACM Symposium on User interface Software and Technology*, pages 139-148. ACM, New York, NY, 2000.

[32] Imrhan, S.N. and Loo, C.H., Trends in finger pinch strength in children, adults, and the elderly. *Human Factors* 31, 6, pages 689-702. Human Factors & Ergonomics Society, Inc., Santa Monica, CA, December 1989.

[33] Inkpen, K.M. 2001. Drag-and-drop versus point-and-click mouse interaction styles for children. *ACM Trans. Comput.-Hum. Interact.* 8, 1, pages 1-33. ACM, New York, NY, March 2001.

[34] Jacob, R.J., Sibert, L.E., McFarlane, D.C., and Mullen, M.P. Integrality and separability of input devices. *ACM Trans. Comput.-Hum. Interact.* 1, 1, pages 3-26. ACM, New York, NY, March 1994.

[35] Jacob, R.J. The future of input devices. *ACM Comput. Surv.* 28, 4es, article 138. ACM, New York, NY, December 1996.

[36] Jacob, R.J., Girouard, A., Hirshfield, L.M., Horn, M.S., Shaer, O., Solovey, E.T., and Zigelbaum, J. Reality-based interaction: a framework for post-WIMP interfaces. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, pages 201-210. ACM, New York, NY, 2008.

[37] Kabbash, P., Buxton, W., and Sellen, A. Two-handed input in a compound task. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 417-423. ACM, New York, NY, 1994.

[38] Kandel, E.R., Schwartz, J.H., and Jessell, T.M. *Principles of Neural Science*, 4th ed., page 433. McGraw-Hill, New York, 2000.

[39] Kim, S., Kim, H., Lee, B., Nam, T., and Lee, W. Inflatable mouse: volume-adjustable mouse with air-pressure-sensitive input and haptic feedback. In *Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems*, pages 211-224. ACM, New York, NY, 2008.

[40] Kruger, R., Carpendale, S., Scott, S.D., and Tang, A. Fluid integration of rotation and translation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 601-610. ACM, New York, NY, 2005.

[41] Li, Y., Hinckley, K., Guan, Z., and Landay, J. A. Experimental analysis of mode switching techniques in pen-based user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 461-470. ACM, New York, NY, 2005.

[42] Liu, J., Pinelle, D., Sallam, S., Subramanian, S., and Gutwin, C. TNT: improved rotation and translation on digital tables. In *Proceedings of Graphics interface 2006*, ACM International Conference Proceeding Series, vol. 137., pages 25-32. Canadian Information Processing Society, Toronto, Ont., Canada, 2006.

[43] Lucas, J. F., Kim, J., and Bowman, D. A. Resizing beyond widgets: object resizing techniques for immersive virtual environments. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, pages 1601-1604. ACM, New York, NY, 2005.

[44] MacKenzie, I. S. and Oniszczak, A. A comparison of three selection techniques for touchpads. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 336-343. ACM Press/Addison-Wesley Publishing Co., New York, NY, 1998.

[45] MacKenzie, I. S., Soukoreff, R. W., and Pal, C. A two-ball mouse affords three degrees of freedom. In *CHI '97 Extended Abstracts on Human Factors in Computing Systems*, pages 303-304. ACM, New York, NY, 1997.

[46] Mandryk, R. L., Rodgers, M. E., and Inkpen, K. M. Sticky widgets: pseudo-haptic widget enhancements for multi-monitor displays. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, pages 1621-1624. ACM, New York, NY, 1995.

[47] Martin, B. and Raisamo, R. TrackMouse: a new solution for 2+2D interactions. In *Proceedings of the Third Nordic Conference on Human-Computer interaction*, pages 89-92. ACM, New York, NY, 2004.

[48] Miller, T. and Zeleznik, R. The design of 3D haptic widgets. In *Proceedings of the 1999 Symposium on interactive 3D Graphics*, pages 97-102. ACM, New York, NY, 1999.

[49] Mizobuchi, S., Terasaki, S., Keski-Jaskari, T., Nousiainen, J., Ryynanen, M., and Silfverberg, M. Making an impression: force-controlled pen input for handheld devices. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, pages 1661-1664. ACM, New York, NY, 2005.

[50] Mountcastle, V.C. *The Sensory Hand: Neural Mechanisms of Somatic Sensation*, page 34. Harvard University Press, 2005.

[51] Paré, M., Behets, C., Cornu, O. Paucity of presumptive ruffini corpuscles in the index finger pad of humans. In *The Journal of Comparitive Neurology*. 456, 3, pages 260-266.Wiley-Liss, Inc., 2003.

[52] Porta, M. and Turina, M. Eye-S: a full-screen input modality for pure eye-based communication. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, pages 27-34. ACM, New York, NY, 2008.

[53] Potter, R. L., Weldon, L. J., and Shneiderman, B. Improving the accuracy of touch screens: an experimental evaluation of three strategies. In *Proceedings of*

the SIGCHI Conference on Human Factors in Computing Systems, pages 27-32. ACM, New York, NY, 1988.

[54] Ramos, G., Boulos, M., and Balakrishnan, R. Pressure widgets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 487-494. ACM, New York, NY, 2004.

[55] Ramos, G. and Balakrishnan, R. Zliding: fluid zooming and sliding for high precision parameter manipulation. In *Proceedings of the 18th Annual ACM Symposium on User interface Software and Technology*, pages 143-152. ACM, New York, NY, 2005.

[56] Ramos, G. A. and Balakrishnan, R. Pressure marks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1375-1384. ACM, New York, NY, 2007.

[57] Rekimoto, J. and Schwesig, C. PreSenseII: bi-directional touch and pressure sensing interactions with tactile feedback. In *CHI '06 Extended Abstracts on Human Factors in Computing Systems*, pages 1253-1258. ACM, New York, NY, 2006.

[58] Ren, X. and Moriya, S. Improving selection performance on pen-based systems: a study of pen-based interaction for selection tasks. *ACM Trans. Comput.-Hum. Interact.* 7, 3, pages 384-416. ACM, New York, NY, September 2000.

[59] Ren, X., Ying, J., Zhao, S., Li, Y. The Adaptive Hybrid Cursor: A Pressure-based Target Selection Technique for Pen-based Interfaces. In *Proceedings of the Interact 2007 Conference*, pages 310-323. 2007.

[60] Sears, A. and Shneiderman, B. High precision touchscreens: design strategies and comparisons with a mouse. *Int. J. Man-Mach. Stud.* 34, 4, pages 593-613. Academic Press Ltd., London, UK, April 1991.

[61] Shi, K., Irani, P., Gustafson, S., and Subramanian, S. PressureFish: a method to improve control of discrete pressure-based input. In *Proceeding of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1295-1298. ACM, New York, NY, 2008.

[62] Shneiderman, B. Direct manipulation: A step beyond programming languages. In *Human-Computer interaction: A Multidisciplinary Approach*, pages 461-467. Morgan Kaufmann Publishers, San Francisco, CA, 1987.

[63] Shoemaker, G. and Gutwin, C. Supporting multi-point interaction in visual workspaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 999-1008. ACM, New York, NY, 2007.

[64] Siio, I., Masui, T., and Fukuchi, K. Real-world interaction using the FieldMouse. In *Proceedings of the 12th Annual ACM Symposium on User interface Software and Technology*, pages 113-119. ACM, New York, NY, 1999.

[65] Srinivasan, P., Birchfield, D., Qian, G., and Kidané, A. A pressure sensing floor for interactive media applications. In *Proceedings of the 2005 ACM SIGCHI international Conference on Advances in Computer Entertainment Technology*, pages 278-281. ACM, New York, NY, 2005.

[66] Srinivasan, M.A. and Chen, J.S. Human performance in controlling normal forces of contact with rigid objects. *Winter Annual Meeting of the American Society of Mechanical Engineers*, vol. 49, pages 119-125. American Society of Mechanical Engineers, January 1993.

[67] Wobbrock, J.O., Chau, D.H., and Myers, B.A. An alternative to push, press, and tap-tap-tap: gesturing on an isometric joystick for mobile phone text entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 667-676. ACM, New York, NY, 2007.

[68] Wobbrock, J.O. and Myers, B.A. EdgeWrite: A new text entry technique designed for stability. In *Proceeding of the RESNA 28th Annual Conference*. RESNA Press, Arlington, VA, 2005.

[69] Zeleznik, R., Miller, T., and Forsberg, A. Pop through mouse button interactions. In *Proceedings of the 14th Annual ACM Symposium on User interface Software and Technology*, pages 195-196. ACM, New York, NY, 2001.

[70] Zhai, S. and Kristensson, P. Shorthand writing on stylus keyboard. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 97-104. ACM, New York, NY, 2003.

[71] Zhai, S., Smith, B. A., and Selker, T. Improving Browsing Performance: A study of four input devices for scrolling and pointing tasks. In *Proceedings of the IFIP Tc13 International Conference on Human-Computer Interaction*, IFIP Conference Proceedings, vol. 96., pages 286-293. Chapman & Hall Ltd., London, UK, 1997.

[72] Zhao, S., Agrawala, M., and Hinckley, K. Zone and polygon menus: using relative position to increase the breadth of multi-stroke marking menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1077-1086. ACM, New York, NY, 2006.

APPENDIX A

QUALITATIVE STUDY QUESTIONNAIRES AND RESULTS

A.1 Consent Form

DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF SASKATCHEWAN
CONSENT FORM

This consent form, a copy of which has been given to you, is only part of the process of informed consent. It should give you the basic idea of what the research is about and what your participation will involve. If you would like more detail about something mentioned here, or information not included here, please ask. Please take the time to read this form carefully and to understand any accompanying information.

Researcher(s): *Sriram Subramanian*, Department of Computer Science (966-4888)
Jared Cechanowicz, Department of Computer Science

Purpose and Procedure:

This study is concerned with investigating the benefits of pressure augmented mouse.

The goal of the research is to determine if users can benefits from adding multiple pressure sensors to different locations in a mouse. We are also investigating various interaction techniques for controlling a pressure augmented mouse.
The session will require 60 minutes, during which you will be asked to carry out several target acquisition tasks.
At the end of the session, you will be given more information about the purpose and goals of the study, and there will be time for you to ask questions about the research.

There is no known risk to you associated with this study.

The data collected from this study will be used in articles for publication in journals and conference proceedings.

As one way of thanking you for your time, we will be pleased to make available to you a summary of the results of this study once they have been compiled (probably in about four weeks). This summary will outline the research and discuss our findings and recommendations. If you would like to receive a copy of this summary, please write down your email address here.

Contact email address____

Confidentiality: All of the information we collect from you (data logged by the computer, observations made by the experimenters, your questionnaire responses and video or audio recording) will be stored so that your name is not associated with it (using an arbitrary participant number). If audio or video recording is used for data collection, the recording will be such that your personal identity is not compromised. Any writeups of the data will not include any information that can be linked directly to you. Please do not put your name or other identifying information on the questionnaire. The research materials will be stored with complete security throughout the entire investigation. Do you have any questions about this aspect of the study?

Right to Withdraw: You are free to withdraw from the study at any time without penalty and without losing any advertised benefits. Withdrawal from the study will not affect your academic status or your access to services at the university. If you withdraw, your data will be deleted from the study and destroyed. If audio or video recording is used for data collection, you have the right to switch off the audio or video recorder at any time during the study. In addition, you are free to not answer specific items or questions on questionnaires.

Your continued participation should be as informed as your initial consent, so you should feel free to ask for clarification or new information throughout your participation. If you have further questions concerning matters related to this research, please contact: Sriram Subramanian (966-4888)

Questions: If you have any questions concerning the study, please feel free to ask at any point; you are also free to contact the researchers at the numbers provided above if you have questions at a later time. This study has been approved on ethical grounds by the University of Saskatchewan Behavioural Sciences Research Ethics Board on (insert date). Any questions regarding your rights as a participant may be addressed to that committee through the Office of Research Services (966-2084). Out of town participants may call collect.

Consent to Participate: *I have read and understood the description provided above; I have been provided with an opportunity to ask questions and my questions have been answered satisfactorily. I consent to participate in the study described above, understanding that I may withdraw this consent at any time. A copy of this consent form has been given to me for my records.*

______________________________ ____________________
(Signature of Participant) (Date)

____
(Signature of Researcher)

A.2 Pressure-Augmented Mouse Experiment 1 TLX and Questionnaire

____

Pressure Mouse - Demographics and TLX Questionnaire

Age: __________

Sex: **M** **F** (circle one)

If you are a student, what is your major? ____________________________

Do you have any visual impairments (including colourblindness)? **Yes** **No**

If so, what? ____

How many hours do you use a computer in a normal week? ________ hours

Of these hours, how many do you spend using: Email? __________

Web Browser? __________

If you play general computer or video games, how frequently do you play them? ________ hours/week

If you do play games, which games or type of games do you play (please list):

____

The following questions correspond to the different interfaces used in the study. If you are unfamiliar with which name is which interface please ask the experimenter.

Pressure Sensor on Left-side of mouse

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Time Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High
Fatigue	Low	|___|___|___|___|___|___|___|	High
Difficulty Selecting High Pressure Targets	Low	|___|___|___|___|___|___|___|	High
Difficulty Selecting Medium Pressure Targets	Low	|___|___|___|___|___|___|___|	High
Difficulty Selecting Low Pressure Targets	Low	|___|___|___|___|___|___|___|	High

1

____

Pressure Sensor on Right-side of Mouse

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Time Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High
Fatigue	Low	|___|___|___|___|___|___|___|	High
Difficulty Selecting High Pressure Targets	Low	|___|___|___|___|___|___|___|	High
Difficulty Selecting Medium Pressure Targets	Low	|___|___|___|___|___|___|___|	High
Difficulty Selecting Low Pressure Targets	Low	|___|___|___|___|___|___|___|	High

Pressure Sensor on Top of mouse

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Time Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High
Fatigue	Low	|___|___|___|___|___|___|___|	High

Pressure Sensor on Top of mouse Cont'd…

____

Difficulty Selecting High Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Medium Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Low Pressure Targets Low |___|___|___|___|___|___|___| High

Click Technique

Mental Demand Low |___|___|___|___|___|___|___| High

Physical Demand Low |___|___|___|___|___|___|___| High

Time Demand Low |___|___|___|___|___|___|___| High

Subjective Performance Low |___|___|___|___|___|___|___| High

Overall Effort Low |___|___|___|___|___|___|___| High

Frustration Low |___|___|___|___|___|___|___| High

Fatigue Low |___|___|___|___|___|___|___| High

Difficulty Selecting High Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Medium Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Low Pressure Targets Low |___|___|___|___|___|___|___| High

Dwell Technique

Mental Demand Low |___|___|___|___|___|___|___| High

Physical Demand Low |___|___|___|___|___|___|___| High

Time Demand Low |___|___|___|___|___|___|___| High

Dwell Technique Cont'd…

____

Subjective Performance Low |___|___|___|___|___|___|___| High

Overall Effort Low |___|___|___|___|___|___|___| High

Frustration Low |___|___|___|___|___|___|___| High

Fatigue Low |___|___|___|___|___|___|___| High

Difficulty Selecting High Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Medium Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Low Pressure Targets Low |___|___|___|___|___|___|___| High

<u>Quick Release Technique</u>

Mental Demand Low |___|___|___|___|___|___|___| High

Physical Demand Low |___|___|___|___|___|___|___| High

Time Demand Low |___|___|___|___|___|___|___| High

Subjective Performance Low |___|___|___|___|___|___|___| High

Overall Effort Low |___|___|___|___|___|___|___| High

Frustration Low |___|___|___|___|___|___|___| High

Fatigue Low |___|___|___|___|___|___|___| High

Difficulty Selecting High Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Medium Pressure Targets Low |___|___|___|___|___|___|___| High

Difficulty Selecting Low Pressure Targets Low |___|___|___|___|___|___|___| High

____

The following questions involve ranking the <u>sensor locations</u> used in the study. Please rank them from 1 to 3 on the following scale: (1: Best, 2: Second Best, 3: Worst)

Please *Rank* the three pressure sensor locations in terms of **Accuracy**

_____ Left _____ Right _____ Top

Please *Rank* the three pressure sensor locations in terms of **Speed**

_____ Left _____ Right _____ Top

Please *Rank* the three pressure sensor locations in terms of **Overall Performance**

_____ Left _____ Right _____ Top

Please *Rank* the three pressure sensor locations in terms of **Ease of Selecting High Pressure Level Targets**

_____ Left _____ Right _____ Top

Please *Rank* the three pressure sensor locations in terms of **Ease of Selecting Medium Pressure Level Targets**

_____ Left _____ Right _____ Top

Please *Rank* the three pressure sensor locations in terms of **Ease of Selecting Low Pressure Level Targets**

_____ Left _____ Right _____ Top

____

The following questions involve ranking the <u>selection techniques</u> used in the study. Please rank them from 1 to 3 on the following scale: (1: Best, 2: Second Best, 3: Worst)

Please *Rank* the three selection techniques in terms of **Accuracy**

_____ Click _____ Dwell _____ Quick-release

Please *Rank* the three selection techniques in terms of **Speed**

_____ Click _____ Dwell _____ Quick-release

Please Rank the three selection techniques in terms of **Overall Performance**

_____ Click _____ Dwell _____ Quick-release

Please *Rank* the three selection techniques in terms of **Ease of Selecting High Pressure Level Targets**

_____ Click _____ Dwell _____ Quick-release

Please *Rank* the three selection techniques in terms of **Ease of Selecting Medium Pressure Level Targets**

_____ Click _____ Dwell _____ Quick-release

Please *Rank* the three selection techniques in terms of **Ease of Selecting Low Pressure Level Targets**

_____ Click _____ Dwell _____ Quick-release

The following are general questions about the study.

Please rate your overall level of fatigue after the study from using the pressure sensors.

Low |___|___|___|___|___|___|___| High

A.3 Pressure-Augmented Mouse Experiment 2 TLX and Questionnaire

____

Prouse Extended - Demographics and TLX Questionnaire

Age: ___________

Sex: **M** **F** (circle one)

If you are a student, what is your major? _______________________________

Do you have any visual impairments (including colour blindness)? **Yes** **No**

If so, what? ____

How many hours do you use a computer in a normal week? _________ hours

Of these hours, how many do you spend using: Email? ___________

Web Browser? ___________

If you play general computer or video games, how frequently do you play them? _________ hours/week

If you do play games, which games or type of games do you play (please list):

____

The following questions correspond to the different selection modes used in the study. If you are unfamiliar with which name is which interface please ask the experimenter.

Normal (1 sensor)

Mental Demand	Low								High
Physical Demand	Low								High
Time Demand	Low								High
Subjective Performance	Low								High
Overall Effort	Low								High
Frustration	Low								High
Fatigue	Low								High

1

____

Assisted Technique (Normal, with assisted selection)

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Time Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High
Fatigue	Low	|___|___|___|___|___|___|___|	High

Refine Technique (Thumb pressure to switch groups)

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Time Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High
Fatigue	Low	|___|___|___|___|___|___|___|	High

____

Tap Technique (Tapping sensors to switch groups)

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Time Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High
Fatigue	Low	|___|___|___|___|___|___|___|	High

The following questions involve ranking the selection modes used in the study. Please rank them from 1 to 4 on the following scale: (1: Best, 2: Second Best, 3: Third Best, 4: Worst)

Please *Rank* the four selection modes in terms of **Accuracy**

_____ **Normal** _____ **Assisted** _____ **Refine** _____ **Tap**

Please *Rank* the four selection modes in terms of **Speed**

_____ **Normal** _____ **Assisted** _____ **Refine** _____ **Tap**

Please *Rank* the four selection modes in terms of **Overall Performance**

_____ **Normal** _____ **Assisted** _____ **Refine** _____ **Tap**

Please *Rank* the four selection modes in terms of **Ease of Selecting Targets in a menu of 4**

_____ **Normal** _____ **Assisted** _____ **Refine** _____ **Tap**

Please *Rank* the four selection modes in terms of **Ease of Selecting Targets in a menu of 12**

_____ **Normal** _____ **Assisted** _____ **Refine** _____ **Tap**

Please *Rank* the four selection modes in terms of **Ease of Selecting Targets in a menu of 16**

_____ **Normal** _____ **Assisted** _____ **Refine** _____ **Tap**

Please *Rank* these 2 selection modes in terms of **Ease of Selecting Targets in a menu of 64**

_____ **Refine** _____ **Tap** (1: Best, 2: Second Best)

____

The following are general questions about the study.

Please rate your overall level of fatigue after the study from using the pressure sensors.

Low |___|___|___|___|___|___|___| High

The following questions correspond to the different selection modes used in the study. Please circle one of the options.

How helpful did you find the Assisted selection technique when you first used the technique? *(Circle one)*

Very Helpful *Helpful* *Not Helpful and Not Distracting* *Distracting* *Very Distracting*

How helpful did you find the Assisted selection technique by the end of the study? *(Circle one)*

Very Helpful *Helpful* *Not Helpful and Not Distracting* *Distracting* *Very Distracting*

A.4 Pressure Buttons Experiment 1 TLX and Questionnaire

____

Buttons - Demographics and TLX Questionnaire

Age: __________ Sex: **M** **F** (circle one)

If you are a student, what is your major? ____________________________

Do you have any visual impairments (including colour blindness)? **Yes** **No**

If so, what? ____

How many hours do you use a computer in a normal week? ________ hours

Of these hours, how many do you spend using: Email? __________ Web Browser? __________

If you play general computer or video games, how frequently do you play them? ________ hours/week

If you do play games, which games or type of games do you play (please list):

____

The following questions correspond to the different input modes used in the study. For each question, please check or draw an 'x' in one of the seven boxes. If you are unfamiliar with which name is which interface please ask the experimenter.

Mouse Button Click

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

Sensor **Click**

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

Sensor **Click** With Audio

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

1

____

Sensor **Tap**

Mental Demand Low |___|___|___|___|___|___|___| High
Physical Demand Low |___|___|___|___|___|___|___| High
Subjective Performance Low |___|___|___|___|___|___|___| High
Overall Effort Low |___|___|___|___|___|___|___| High
Frustration Low |___|___|___|___|___|___|___| High

Sensor **Tap** With Audio

Mental Demand Low |___|___|___|___|___|___|___| High
Physical Demand Low |___|___|___|___|___|___|___| High
Subjective Performance Low |___|___|___|___|___|___|___| High
Overall Effort Low |___|___|___|___|___|___|___| High
Frustration Low |___|___|___|___|___|___|___| High

The following questions involve ranking the input modes used in the study. Please rank them from 1 to 5 on the following scale: (1: Best, 2: Second Best, 3: Third Best, 4: Fourth Best, 5: Worst)

Please *Rank* the four input modes in terms of **Overall Effort**

_____ **Button Click** _____ **Sensor Click** _____ **Sensor Click (Audio)** _____ **Tap** _____ **Tap (Audio)**

Please *Rank* the four input modes in terms of **Speed**

_____ **Button Click** _____ **Sensor Click** _____ **Sensor Click (Audio)** _____ **Tap** _____ **Tap (Audio)**

Please *Rank* the four input modes in terms of **Overall Performance**

_____ **Button Click** _____ **Sensor Click** _____ **Sensor Click (Audio)** _____ **Tap** _____ **Tap (Audio)**

Please *Rank* the four input modes in terms of **Frustration**

_____ **Button Click** _____ **Sensor Click** _____ **Sensor Click (Audio)** _____ **Tap** _____ **Tap (Audio)**

#________

The following questions correspond to the different input modes used in the study. Please circle one of the options.

How helpful did you find the audible “clicks” for the Sensor **Click** with Audio mode when you first used the technique? *(Circle one)*

Very Helpful *Helpful* *Not Helpful and Not Distracting* *Distracting* *Very Distracting*

How helpful did you find the audible “clicks” for the Sensor **Click** with Audio mode by the end of the study? *(Circle one)*

Very Helpful *Helpful* *Not Helpful and Not Distracting* *Distracting* *Very Distracting*

How helpful did you find the audible “clicks” for the Sensor **Tap** with Audio mode when you first used the technique? *(Circle one)*

Very Helpful *Helpful* *Not Helpful and Not Distracting* *Distracting* *Very Distracting*

How helpful did you find the audible “clicks” for the Sensor **Tap** with Audio mode by the end of the study? *(Circle one)*

Very Helpful *Helpful* *Not Helpful and Not Distracting* *Distracting* *Very Distracting*

A.5 Pressure Buttons Experiment 2 TLX and Questionnaire

____

Buttons 2 - Demographics and TLX Questionnaire

Age: __________ Sex: **M** **F** (circle one)

If you are a student, what is your major? ______________________________

Do you have any visual impairments (including colour blindness)? **Yes** **No**

If so, what? ____

How many hours do you use a computer in a normal week? ________ hours

Of these hours, how many do you spend using: Email? __________ Web Browser? __________

If you play general computer or video games, how frequently do you play them? ________ hours/week

If you do play games, which games or type of games do you play (please list):

____

The following questions correspond to the different input modes used in the study. For each question, please check or draw an 'x' in one of the seven boxes. If you are unfamiliar with which name is which interface please ask the experimenter.

Mouse Button Click

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

Sensor **Click**

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

Sensor **Tap**

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

1

____

HardPress

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

HardPress With Audio

Mental Demand	Low	|___|___|___|___|___|___|___|	High
Physical Demand	Low	|___|___|___|___|___|___|___|	High
Subjective Performance	Low	|___|___|___|___|___|___|___|	High
Overall Effort	Low	|___|___|___|___|___|___|___|	High
Frustration	Low	|___|___|___|___|___|___|___|	High

The following questions involve ranking the input modes used in the study. Please rank them from 1 to 5 on the following scale: (1: Best, 2: Second Best, 3: Third Best, 4: Fourth Best, 5: Worst)

Please *Rank* the four input modes in terms of **Overall Effort**

_____ **Button Click** _____ **Sensor Click** _____ **Tap** _____ **HardPress** _____ **HardPress (Audio)**

Please *Rank* the four input modes in terms of **Speed**

_____ **Button Click** _____ **Sensor Click** _____ **Tap** _____ **HardPress** _____ **HardPress (Audio)**

Please *Rank* the four input modes in terms of **Overall Performance**

_____ **Button Click** _____ **Sensor Click** _____ **Tap** _____ **HardPress** _____ **HardPress (Audio)**

Please *Rank* the four input modes in terms of **Frustration**

_____ **Button Click** _____ **Sensor Click** _____ **Tap** _____ **HardPress** _____ **HardPress (Audio)**

____

The following questions correspond to the different input modes used in the study. Please circle one of the options.

How helpful did you find the audible "clicks" for the HardPress with Audio mode when you first used the technique? *(Circle one)*

Very Helpful   *Helpful*   *Not Helpful and Not Distracting*   *Distracting*   *Very Distracting*

How helpful did you find the audible "clicks" for the HardPress with Audio mode by the end of the study? *(Circle one)*

Very Helpful   *Helpful*   *Not Helpful and Not Distracting*   *Distracting*   *Very Distracting*

A.6 Augmented Interactions TLX and Questionnaire

____

Expressive Widgets - Demographics and TLX Questionnaire

Age: __________ Sex: **M** **F** (circle one)

If you are a student, what is your major? ______________________________

Do you have any visual impairments (including colour blindness)? **Yes** **No**

If so, what? ____

How many hours do you use a computer in a normal week? ________ hours

Of these hours, how many do you spend using: Document editor? _______ Web Browser? _______ File Explorer? _____

If you play general computer or video games, how frequently do you play them? ________ hours/week

If you do play games, which games or type of games do you play (please list):

____

If you have used any of the following devices for computing or gaming please circle them.

Footpedal Slider Joystick Iso-Joystick (the nub on keyboards and mice) Dial/Knob Steering Wheel

Stylus (PDA/tablet) Multitouch screen (iPhone or similar) Xbox 360 controller PS2 controller Wii

The following questions correspond to the different widgets from the study. For each question, please circle the response you most agree with. If you are unfamiliar with which name is which widget please ask the experimenter.

Scroll Buttons

How difficult/easy was it to ***learn*** how to use these new pressure sensitive scroll buttons?

very difficult *difficult* *not easy & not difficult* *easy* *very easy*

How easy/difficult was it to ***use*** these new pressure sensitive scroll buttons after you learned to use them?

very difficult *difficult* *not easy & not difficult* *easy* *very easy*

Compared to standard scroll buttons, how much less/more efficiently would you perform tasks with the new buttons?

much less efficiently *less efficiently* *not more & not less efficiently* *more efficiently* *much more efficiently*

How poorly/well suited was the mapping of pressure to scroll speed?

Very Poorly Suited *Poorly Suited* *Not well & not poorly Suited* *Well Suited* *Very Well Suited*

How undesirable/desirable is having this new widget and the interactions that it allows?

very undesirable *undesirable* *not undesirable & not desirable* *desirable* *very desirable*

Which type of scroll buttons do you prefer overall?

Pressure sensitive *Standard* *No Preference*

In what situations, if any, would you like to have pressure sensitive scroll buttons?

____

____

____

In what situations would the pressure sensitive scroll buttons make a task more difficult/slower?

____

____

____

1

____

Scroll Buttons Continued...

If you could change anything about the way this new widget works, what would you change?

____

____

____

Any other comments:

____

____

____

____

Scroll Thumb

How difficult/easy was it to ***learn*** how to use this new pressure sensitive scroll thumb?

very difficult *difficult* *not easy & not difficult* *easy* *very easy*

How easy/difficult was it to ***use*** the new pressure sensitive scroll thumb after you learned to use it?

very difficult *difficult* *not easy & not difficult* *easy* *very easy*

Compared to a standard scroll thumb, how much less/more efficiently would you perform tasks with the new thumb?

much less efficiently *less efficiently* *not more & not less efficiently* *more efficiently* *much more efficiently*

How poorly/well suited was the mapping of pressure to zoom-out distance?

Very Poorly Suited *Poorly Suited* *Not well & not poorly Suited* *Well Suited* *Very Well Suited*

How undesirable/desirable is having this new widget and the interactions that it allows?

very undesirable *undesirable* *not undesirable & not desirable* *desirable* *very desirable*

Which type of scroll thumb do you prefer overall?

Pressure sensitive *Standard* *No Preference*

In what situations, if any, would you like to have this new scroll thumb?

____

____

____

In what situations would the pressure sensitive scroll thumb make a task more difficult/slower?

____

____

____

If you could change anything about the way this new widget works, what would you change?

____

____

____

____

Scroll Thumb Continued...

Any other comments:

____
____
____
____

Web Link "Peaking"

How difficult/easy was it to ***learn*** how to use these new pressure sensitive web links?

very difficult   difficult   not easy & not difficult   easy   very easy

How easy/difficult was it to ***use*** the new pressure sensitive web links after you learned to use them?

very difficult   difficult   not easy & not difficult   easy   very easy

Compared to standard web links, how much less/more efficiently would you perform tasks with the new links?

much less efficiently   less efficiently   not more & not less efficiently   more efficiently   much more efficiently

How poorly/well suited was the mapping of pressure to preview thumbnail size?

Very Poorly Suited   Poorly Suited   Not well & not poorly Suited   Well Suited   Very Well Suited

How undesirable/desirable is having this new widget and the interactions that it allows?

very undesirable   undesirable   not undesirable & not desirable   desirable   very desirable

Which type of links do you prefer overall?

Pressure sensitive   Standard   No Preference

In what situations, if any, would you like to have these new pressure sensitive web links?

____
____
____

In what situations would the pressure sensitive web links make a task more difficult/slower?

____
____
____

If you could change anything about the way this new widget works, what would you change?

____
____
____

Any other comments:

____
____
____
____

____

Back / Forward Buttons

How difficult/easy was it to ***learn*** how to use these new pressure sensitive back / forward buttons?

very difficult   difficult   not easy & not difficult   easy   very easy

How easy/difficult was it to ***use*** the new pressure sensitive back / forward buttons after you learned to use them?

very difficult   difficult   not easy & not difficult   easy   very easy

Compared to the standard buttons, how much less/more efficiently would you perform tasks with these new buttons?

much less efficiently   less efficiently   not more & not less efficiently   more efficiently   much more efficiently

How poorly/well suited was the mapping of pressure to back / forward distance?

Very Poorly Suited   Poorly Suited   Not well & not poorly Suited   Well Suited   Very Well Suited

How undesirable/desirable is having this new widget and the interactions that it allows?

very undesirable   undesirable   not undesirable & not desirable   desirable   very desirable

Which type of back / forward buttons do you prefer overall?

Pressure sensitive   Standard   No Preference

In what situations, if any, would you like to have these new pressure sensitive back / forward buttons?

____

____

____

In what situations would the pressure sensitive back / forward buttons make a task more difficult/slower?

____

____

____

If you could change anything about the way this new widget works, what would you change?

____

____

____

Any other comments:

____

____

____

____

Folder Security

How difficult/easy was it to ***learn*** how to use this new pressure sensitive folder security?

very difficult   difficult   not easy & not difficult   easy   very easy

How easy/difficult was it to ***use*** the new pressure sensitive folder security after you learned to use it?

very difficult   difficult   not easy & not difficult   easy   very easy

____

Folder Security Continued...

Compared to confirming with a dialog box, how much less/more efficiently would you perform tasks with pressure sensitive security?

much less efficiently *less efficiently* *not more & not less efficiently* *more efficiently* *much more efficiently*

How poorly/well suited was the mapping of pressure to confidence?

Very Poorly Suited *Poorly Suited* *Not well & not poorly Suited* *Well Suited* *Very Well Suited*

How undesirable/desirable is having this new widget and the interactions that it allows?

very undesirable *undesirable* *not undesirable & not desirable* *desirable* *very desirable*

Which type of folder security do you prefer overall?

Pressure sensitive *Standard* *No Preference*

In what situations, if any, would you like to have this new pressure sensitive security?

In what situations would pressure sensitive security make a task more difficult/slower?

If you could change anything about the way this new widget works, what would you change?

Any other comments:

Folder and File Types

How difficult/easy was it to ***learn*** how to use this new pressure sensitive folder and file sorting?

very difficult *difficult* *not easy & not difficult* *easy* *very easy*

How easy/difficult was it to ***use*** the new pressure sensitive folders with file sorting after you learned to use it?

very difficult *difficult* *not easy & not difficult* *easy* *very easy*

Compared to confirming with a drag action with a dialog box, how much less/more efficiently would you perform tasks with pressure sensitive file sorting?

much less efficiently *less efficiently* *not more & not less efficiently* *more efficiently* *much more efficiently*

How poorly/well suited was the mapping of pressure to confidence?

Very Poorly Suited *Poorly Suited* *Not well & not poorly Suited* *Well Suited* *Very Well Suited*

____

Folder and File Types Continued...

How undesirable/desirable is having this new widget and the interactions that it allows?

very undesirable   undesirable   not undesirable & not desirable   desirable   very desirable

Which type of sorting and confirming actions do you prefer overall?

Pressure sensitive   Standard   No Preference

In what situations, if any, would you like to have this new pressure sensitive confirming and file sorting?

In what situations would pressure sensitive confirming and sorting make a task more difficult/slower?

If you could change anything about the way this new widget works, what would you change?

Any other comments:

The following are general questions about the entire study. If you have any questions please ask the experimenter.

Overall, how difficult/easy was it to ***learn*** how to use these new widgets?

very difficult   difficult   not easy & not difficult   easy   very easy

Overall, how easy/difficult was it to ***use*** the new widgets after you learned to use them?

very difficult   difficult   not easy & not difficult   easy   very easy

How undesirable/desirable is having new widgets like these?

very undesirable   undesirable   not undesirable & not desirable   desirable   very desirable

Overall, which type of widgets would you prefer to have for your day to day computing?

Pressure sensitive   Standard   No Preference

A.7 Augmented Interactions Questionnaire Results

	1	2	3	4	5	6	7	8	9	10
Scroll Buttons										
Ease of Learning	5	5	4	4	4	5	5	5	5	4
Ease of Use	5	5	4	3	2	5	5	4	5	4
Efficiency Compared to Standard Widget	5	4	4	2	4	4	4	3	4	4
How Suited to Pressure?	5	4	3	3	4	3	3	3	4	3
Desirability of New Functionality	4	4	4	2	4	5	5	4	4	3
Preference (1=new, -1=old)	1	1	0	-1	1	-1	1	-1	0	0
Scroll Thumb										
Ease of Learning	5	5	4	5	4	5	5	5	5	4
Ease of Use	4	5	4	4	2	5	5	5	5	4
Efficiency Compared to Standard Widget	4	5	5	3	4	4	4	4	4	4
How Suited to Pressure?	3	5	4	4	3	4	3	4	5	4
Desirability of New Functionality	3	5	5	3	4	5	4	4	5	4
Preference (1=new, -1=old)	0	1	1	-1	1	1	1	0	1	1
Peaking										
Ease of Learning	4	5	4	3	4	5	5	5	5	4
Ease of Use	5	5	4	2	2	5	4	5	5	4
Efficiency Compared to Standard Widget	3	4	4	1	3		3	4	4	4
How Suited to Pressure?	3	5	2	2	4	3	3	4	4	3
Desirability of New Functionality	4	4	4	2	3	4	2	5	3	4
Preference (1=new, -1=old)	1	1	1	-1	0	1	-1	1	0	1
Back/Forward Buttons										
Ease of Learning	5	5	4	4	4	4	5	5	5	4
Ease of Use	5	4	2	4	1	5	4	3	3	3
Efficiency Compared to Standard Widget	5	5	3	3	3	2	4	2	4	4
How Suited to Pressure?	5	5	2	3	3	3	4	3	3	3
Desirability of New Functionality	5	5	3	3	3	3	4	3	4	4
Preference (1=new, -1=old)	1	1	-1	-1	0	-1	1	-1	1	1
Security										
Ease of Learning	5	5	5	4	4	5	5	5	5	4
Ease of Use	5	5	5	4	4	5	5	5	5	4
Efficiency Compared to Standard Widget	5	4	5	3	4	4	5	5	5	4
How Suited to Pressure?	5	5	4	3	3	3	2	4	5	4
Desirability of New Functionality	5	5	5	3	3	4	5	5	5	3
Preference (1=new, -1=old)	1	1	1	-1	0	1	1	1	1	0
File Types										
Ease of Learning	5	5	4	5	4	5	5	5	5	4
Ease of Use	5	5	4	5	3	5	5	5	5	4
Efficiency Compared to Standard Widget	5	4	3	4	4	3	5	5	5	4
How Suited to Pressure?	5	5	3	3	3	3	4	4	4	4
Desirability of New Functionality	5	4	3	4	3	3	5	5	4	4
Preference (1=new, -1=old)	1	1	0	1	-1	-1	1	1	1	1
OVERALL										
Ease of Learning	5	5	4	4	4	5	5	5	5	4
Ease of Use	5	5	4	4	2	5	5	5	5	4
Desirability of New Functionality	4	5	4	3	4	4	5	4	4	4
Preference (1=new, -1=old)	1	1	1	-1	1	0	1	0	1	0