Mode Selection Techniques for Pen Input Systems

Keith Deming and Edward Lank
Computer Science Department
San Francisco State University
1600 Holloway Avenue, San Francisco, CA, 94132
kademing@sfsu.edu, lank@cs.sfsu.edu

ABSTRACT
In this paper we introduce a variation on pen-input systems that uses pen orientation to enter selection mode. We compare the use of pen orientation for mode selection to a system which uses the Inferred-Mode protocol as described by Saund and Lank [7]. Our goals are better understand how users in general expect and prefer to interact with pen input systems and how user performance is altered through the selective use of intelligence in pen interfaces.

Author Keywords
Pen interaction, stylus orientation, mode, Inferred Mode, sketch, DWIM.

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

INTRODUCTION

While pen-based interaction offers a number of advantages over traditional keyboard and mouse interfaces in the areas or problem solving and data entry, it also currently suffers from some significant drawbacks. Authors have highlighted the advantages of pen-based interaction with a computer for domains as diverse as problem solving, note taking, sketching, and mobile data entry. The typical advantages noted by authors include:

- The ability to input data without requiring a desk on which to set up and carefully type information.
- The informal nature of the information entered, thus enabling discovery tasks.
- The ability to enter diagrammatic data more rapidly.

Unfortunately, while pen interaction continues to be advocated, its adoption has not been particularly smooth. We hypothesize that one reason for this failure of adoption is a result of the typical failures of user interfaces, namely the need to manipulate both the state of the program and the information content of the program [5]. Researchers have noted that even light-weight constraints on users serve to distract users from their primary task and force them [1]. This is a particular impediment to the usability of pen-based computer interfaces, where users expect the ease of use of pen and paper interfaces, but the computer rarely delivers.

Recently, Saund and Lank introduced the inferred mode protocol in an attempt to alleviate the mode problem in pen-based interfaces [7]. Their hypothesis was that mode-based interaction was one stumbling block in the adoption of pen-based interfaces, or, more specifically, that mode errors were a common occurrence that distracted the user from their primary task and demanded, from the user, that attention be allocated to the user interface. However, in Saund and Lank's work, little validation was performed.

In this paper, we describe our work in validating the inferred mode protocol. We have designed a sketching application using the Microsoft Tablet PC SDK that replicates the behavior of the InkScribe system [7], and a Tablet PC interface that uses stylus orientation as a tactile mode cue for users. In this paper, we describe the testing of these interfaces with the goal of understanding both the attraction and pitfalls of DWIM-based interaction for pen interfaces.

This paper is organized as follows. We first outline some related work in tactile or quasi-mode interfaces. We also describe the inferred mode protocol as outlined by Saund and Lank. Next, we describe our sketch applications. We then describe a user trial which evaluates both speed of input and user satisfaction with both the inferred mode protocol and stylus orientation as techniques for mode selection. Finally, we describe our on-going work to understand the use of intelligence in pen-based interaction.

BACKGROUND

One challenge faced by pen-computer interfaces for drawing applications is the need to overload the limited input modality of the electronic stylus to provide both the ability to manipulate the content in a sketch and the ability to interact with user interface to control the interpretation of any gesture preformed by the user. One example of this is the need to provide both the ability to draw and to edit in interfaces. To overload the limited input modality of a typical stylus, interface designers have resorted to a variety of techniques including pressing a button on the barrel of the stylus, holding the stylus stationary while pressing it against the display for a time-out period, or pressing a button in the toolbar. Saund and Lank note problems with these interaction techniques [7].

Several alternatives have been explored by researchers to overcome the drawbacks of software modes in user interfaces, including clear depiction of mode [6], gesture-based interaction [2], or tactile feedback [3, 4]. Our interest is in comparing the use of tactile feedback, specifically quasi-modes with DWIM-based interaction, specifically the Inferred Mode Protocol [7] to determine how effective each interaction method is in simple sketching interfaces and to begin to understand the advantages and pitfalls of computational intelligence in pen interaction.

Tactile feedback has been used to control selection mode in pen interfaces, specifically in the Flatland system [4]. In this system, designed to run on a touch sensitive *SmartBoard* marketed by Smart Technologies, a button on the electronic pen tray was used to switch from drawing mode (when not pressed) to selection mode (when pressed). As well, most stylus systems come with a double ended stylus where one end is used to mark and the other to erase.

Figure 1: The reasoning diagram for the Inferred Mode Protocol.

In all the above systems, users control the software mode, either by switching modes or by holding the software in some quasi-mode. In contrast, Saund and Lank propose the use of computational analysis to determine the most likely interpretation of a user's action, in other words, a *Do what I mean*, or DWIM, interface [5]. The system performs an action if the appropriate action can be determined unambiguously. If not, the system resorts to the use of a mediator.

Saund and Lank call their interaction technique the *Inferred Mode Protocol* [7]. The interpretation of a user's action is performed by a simple decision tree which analyzes both user action and context using the reasoning diagram pictured in Figure 1.

SIMPLE SKETCHING INTERFACES

Our overall goal is to evaluate techniques for addressing the mode problem. To address mode problems in interfaces, we choose to focus on tactile cues and on DWIM-based interfaces.

We have created two simplified sketching interfaces. The first uses the inferred mode protocol to switch between drawing and inking modes. The second uses the orientation of the stylus to determine modes. When the stylus is inverted, it is in *command* mode, and any gesture is interpreted as selection. The sketch program we created is pictured in Figure 2.

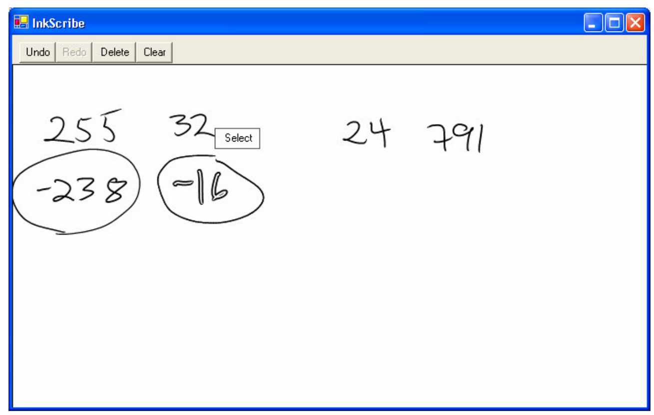

Figure 2: The simplified sketch application used to test techniques for overcoming mode errors.

USER STUDY

We asked 16 users to complete a simple task using both stylus interfaces. Eight users started with the inferred mode protocol application (InkScribe), and the other eight started with the stylus orientation interface (StylScribe).

The sample task is pictured in Figure 2. Users were given a list of seven numbers. They were asked to transcribe the numbers and then order the numbers using a *selection sort* algorithm. Specifically, once they had transcribed the numbers, they were to select the smallest number, move it into the first position on the row below, then circle it. They were to repeat this for subsequent numbers until the list was sorted and each number in the list was circled. We included both positive and negative numbers in our list to add a bit more cognitive load. A stopwatch application on the computer allowed users to time themselves.

Users were given four identical lists of numbers. The lists were always entered in the same order. The first list was used to allow the user to practice performing the task with the first sketch program, and the second list was used in the timed task. The third list was used to practice working with the second interface, and the fourth list was the timed task using the second interface.

After the trial, we administered a post-trial questionnaire that asked users to rate on a scale of 1=easy to 10=hard how easy or hard each of the interface techniques were to use. We also asked users which interaction technique they liked best.

RESULTS

The self-timing information gathered from our user trial seems to indicate that users are better able to interact with the StylScribe system. As can be seen from the data in Table 1, not only was the average time to complete tasks in StylScribe less than that in InkScribe, there seems to be a much lesser variation from one user to the next with regard to performance.

	StylScribe	InkScribe
Average Time to Complete Tasks	55.24	59.41
Standard Deviation	11.31	18.66

Table 1: Summary of timing information gathered from all subjects.

When asked to rate the two systems on ease of use, most users found StylScribe easier to use than InkScribe. Some users noted that, even though StylScribe could be more cumbersome, with practice it became more “intuitive” to use than InkScribe. Figure 3 gives the results of user rating, and seems to indicate that StylScribe was slightly easier to use than InkScribe. Users noted that sometimes the mediator was difficult to target, or that it would sometimes not appear when they expected it would.

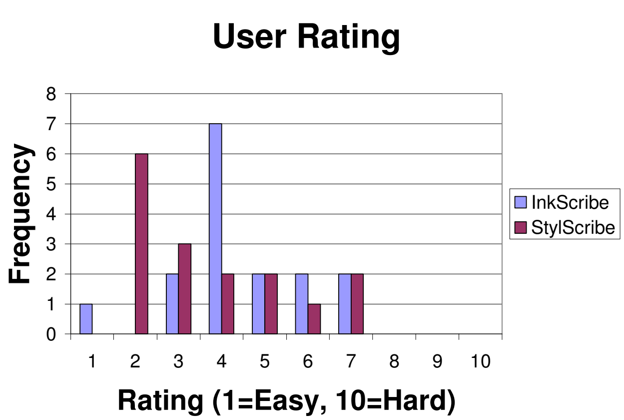

Figure 3: Summary of user ratings for both systems tested with regard to ease of use.

Our final question asked users to express their overall preference for one system or the other. Figure 4 shows a preference for StylScribe over InkScribe by a margin of over 2-to-1. It may be interesting to note however that two users who expressed preference for the StylScribe software rated StylScribe more difficult to use than InkScribe. One of these participants preferred StylScribe because it seemed more “natural,” and the other preferred StylScribe because “the stylus flipping version has the potential to simplify the process.

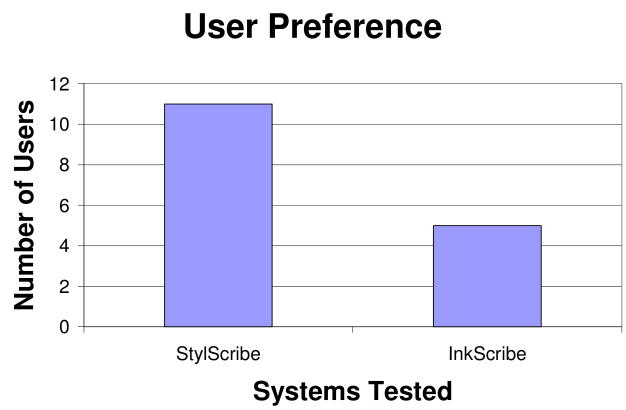

Figure 4: Response to the question, "Which program did you like best?"

DISCUSSION

Nielsen noted, in 1993, that the next generation of user interfaces would incorporate what he termed *noncommand* styles of interaction, where users need not specify commands to interact with the system [5]. While research has progressed in areas such as gesture recognition, there is still a command-based flavor to many of our interactions.

In many pen-based applications, this command nature is particularly apparent. Consider the common note taking applications OneNote and Windows Journal. In both these interfaces, there is a need to issue commands to the system via the use of buttons at the top of the display. The user can select whether any gesture will be the last used pen, an eraser, or keyboard input via typing. Even the use of a button on the barrel of the stylus, or the eraser tip, constitutes command-based interaction, as distinct from Nielsen’s noncommand interaction.

Nielsen notes that, in a noncommand interface, the computer role is one of “interpreting user’s actions and doing what it deems appropriate” [5], and that interface control is relegated to the computer. Clearly, the inferred mode protocol is one instance of a noncommand interaction technique.

What seems most interesting from our user trial is that users seem to prefer StylScribe even when the preference is based solely on “feel.” They note that they find stylus flipping "more natural" or that, although it took longer, it was "cleaner" in some way to slip the stylus. Finally, the separation of "movement vs. writing" seemed to allow them to feel more in control of the interaction.

We note that software can always be improved. A bigger "Select" button might help, as well as a better test for enclosure. The need to occasionally undo and redo in inferred mode protocol often caused users to note errors with the software, not the interaction technique.

Finally, and qualitatively, we note that as users become expert with the inferred mode protocol, it does tend to be faster than either stylus flipping or a standard mode button interface (an unsurprising result given that KLM anlaysis would indicate that mode-based interaction would be slightly slower even if the select button was always necessary to select). As users become more experienced with systems, their preferences might change to the interface that allows more rapid drawing (ours certainly has).

CONCLUSIONS AND FUTURE WORK

In this paper, we describe a user trial that compares the Inferred Mode Protocol described by Saund and Lank [7] with a selection mechanism based on stylus inversion. When asked to rate the interaction techniques, users preferred, by a two to one margin, the use of stylus inversion over the use of the inferred mode protocol, stating that stylus inversion seemed more natural.

As a next step in our work, we are implementing a Windows Journal like interface implementing the inferred mode protocol. We plan to provide students with this interface as an alternative to the standard windows application for classroom-based note taking and compare their preferences over longer periods of time. As well, we plan a more careful analysis of the distaste users feel for DWIM-based interaction with the goal of understanding when, how, and how much intelligence can be incorporated into interfaces in stylus systems.

REFERENCES

1. Arvo, J. Computer Aided Serendipity. Proc. Graphics Interface '99, 1999, 183-192.
2. Kurtenbach, G. and Buxton, W. Issues in Combining Marking and Direct Manipulation Techniques. *Proc. UIST'91,* ACM Press (1991) 137 – 144.
3. Mohamed, F. and Fels, S. LMNKui: Overlaying computer controls on a piano controller keyboard. *Proc. CHI 2002*, ACM Press (2002), 638-639.
4. Mynatt, E., Igarashi, T. Edwards, W.K. and LaMarca, A. Flatland: New Dimensions in Office Whiteboards. *Proc. CHI'99*, ACM Press (1999), 346 - 353.
5. Nielsen, J. Noncommand User Interfaces. *CACM 36,* 4 (1993), 83 – 99.
6. Ryu, H. Will it be upper-case or will it be lower-case: can a prompt for text be a mode signal. *Proc. CHI 2002*, ACM Press (2002), 836- 837.
7. Saund, E and Lank, E. Stylus Input and Editing Without Prior Selection of Mode. *Proc. UIST 2003*, ACM Press (2002), 213 – 216.