EUROGRAPHICS Workshop on Sketch-Based Interfaces and Modeling (2008)
C. Alvarado and M.- P. Cani (Editors)

An Empirical Study in Pen-Centric User Interfaces: Diagramming

Andrew S. Forsberg[1], Andrew Bragdon[1], Joseph J. LaViola Jr.[2], Sashi Raghupathy[3], Robert C. Zeleznik[1]

[1]Brown University, Providence, RI, USA
[2]University of Central Florida, Orlando, FL, USA
[3]Microsoft Corporation, Redmond, WA, USA

Abstract

We present a user study aimed at helping understand the applicability of pen-computing in desktop environments. The study applied three mouse-and-keyboard-based and three pen-based interaction techniques to six variations of a diagramming task. We ran 18 subjects from a general population and the key finding was that while the mouse and keyboard techniques generally were comparable or faster than the pen techniques, subjects ranked pen techniques higher and enjoyed them more. Our contribution is the results from a formal user study that suggests there is a broader applicability and subjective preference for pen user interfaces than the niche PDA and mobile market they currently serve.

Categories and Subject Descriptors (according to ACM CCS): H.5.2 [User Interfaces]: Evaluation/Methodology

1. Introduction

Research on pen computing can be traced back at least to the early 60's. Curiously though, there is little formal understanding of when, where, and for whom pen computing is the user interface of choice. Certainly there seems to be a clear benefit for pen-computing for users who are not sitting down, although even that notion is challenged by pocket-sized devices like the iPhone which rely on finger instead of stylus input. Within a small niche somewhere between mobile and desktop, pen-computing is emerging as a de facto standard for electronic notebooks and free-form note-taking. However, in the entrenched domain of desktop environments, pen-computing for some reason has not been able to supplant mouse and keyboard user interfaces, even for tasks that seem ideal for pen-input, such as diagramming.

The focus of this paper, therefore, is to shed light on whether the slow adoption rate in desktop environments of pen-computing, even for nominally pen-centric tasks, reflects a fundamental misunderstanding of the value of pen computing or whether real benefits exist but are being overlooked as a result of external factors. In essence we are challenging the assumption that, just because a task superficially appears pen-centric, users will in fact derive a significant benefit from using a pen-based interface. Our approach is to quantify formally, through head-to-head evaluation, user performance and relative preference for a representative sampling of both keyboard and mouse, and pen-based input techniques. In this context, we base our experiments on the null hypothesis that "pen-centric input affords no significant advantage to keyboard and mouse input even for a 2D diagramming task." The alternative hypothesis is that pen-based user interfaces for some tasks can, in fact, be superior to the best keyboard and mouse user interfaces.

While this study was conducted in the context of 2D diagramming, note that the simple, fundamental user actions and abstractions of the tasks (i.e., pointing, dragging, writing text) occur frequently in general graphical user interface tasks like document creation and editing, desktop management, web-browsing, and email. Thus the results should be considered beyond diagramming alone and shed light on whether pen-input is really a niche application area.

To maximize the generality of our results, we strategically chose tasks that we felt were the most representative instead of tasks that were tailored specifically to the null hypothesis. For example, we likely could disprove the null hypothesis by focusing on diagramming tasks that involve free-form organic curves or a large set of 2D shapes, but these tasks would only be of interest to a relatively narrow user-base of skilled artists (who in many cases already use pen-input). Similarly, we believe that we could artificially support the null hypothesis by testing pen-centric input under conditions where recognition accuracy would likely be low, but again avoided these conditions because they would limit the results to an understanding of the impact of the current state-of-the-art in recognition algorithms and not a fundamental understanding of pen-centric input.

Instead, the study we conducted considered a representative set of simple diagramming tasks involving only three primitives: boxes, lines, and text. Further, the tasks were designed such that we could guarantee 100% recognition accuracy and thus avoid the confounding issues of error rate and choice of error correction techniques. However, even in this limited context, multiple keyboard-and-mouse and pen input user interfaces are possible, and so our test involves three different keyboard and mouse interfaces and three different pen-centric interfaces, each optimized to one of three criteria: self-disclosure, directness, or performance. In a pilot study we found none of the techniques were inappropriate for diagramming.

It is important to stress that simple diagramming tasks do not imply simple user interactions. Two additional reasons simple diagrams are important to investigate are more people produce simple diagrams than complex ones (i.e., there can be a broad impact), and from anecdotal information gathering we observe that even for simple diagrams there is a need for better diagramming tools.

Although our tasks have been designed around a set of artificial diagramming tasks, we believe that the results are more general. In particular, the tasks allow us to isolate the *interaction parameters* which we are much more interested in than the specific tasks and their qualities. We also believe that this study sheds light outside of diagramming on common assumptions about pen-centric input, including:

- gestures are hard to learn.
- intuitive input (i.e., what people would do with pencil and paper) is more desirable than gestural input.
- entering text with a pen is harder than with a keyboard.
- discoverability is valued more than performance.
- sketching is much easier to do with a pen than keyboard and mouse.
- transition time between devices and/or modes is minimized through pen-input

Ultimately, we envision this study as a basis toward a broad understanding of the applicability of pen-computing in desktop environments. If pen-computing can be shown to have a significant benefit for diagramming tasks, then follow-up studies are justified to quantify the benefit of pen input for other desktop tasks, including those that are less intuitively pen-centric.

2. Related Work

There has been almost 40 years of graphical user interface (GUI) research and development leading to today's conventional keyboard and mouse interfaces. Pen research and development also has a long history dating back to the 1960's. Much research has been done, in particular, to develop recognition and pen-based user interface techniques–e.g., [Dav02] [SL03] [LM01] [ZM06].

Our study has subjects perform a copy-a-target-diagram task and compares head-to-head both pen- and mouse-based techniques against each other. Apte compared pen and mouse input for text-free diagram entry and editing, and found the pen was twice as fast as a mouse for more complex diagrams than we used [AK93]. Our study included text which is integral to most diagrams, and we also systematically explored more variations of pen- and mouse-input techniques. [SSD01] compared their drawing recognition technique to XFig, a Unix toolbar-based system for creating diagrams, but did not compare it with systems using keyboard modifiers or gestural interaction. [AD04] developed a system that recognizes diagrams, but did not compare their system to non-drawing interfaces.

MacKenzie et al. compared mouse, trackball, and pen input for pointing and dragging tasks and found the highest index of performance was for the pen during pointing and for the mouse during dragging [MSB91]. Torres evaluated a novel straight-line-only, gestural interface designed for novice older users and his pen interface based on gestural input had a higher usability measure than a mouse-based interface for older people. [Tor06] This study did not compare shortcut techniques and a literal drawing technique was not applicable because of the domain. Apitz et al.'s drawing application CrossY accepts both pen and mouse input, but does not support primitives like straight lines, rectangles, or text. CrossY has not been formally evaluated [AG04].

Also related are studies of GUI and keyboard shortcut interactions such as [LNPS05] who found only a few out of 251 experienced Microsoft Word users transitioned from GUI to keyboard shortcuts, but also that if they did they would have been more efficient. To the best of our knowledge no study quite like ours has been run.

3. Methods

3.1. Subjects and Apparatus

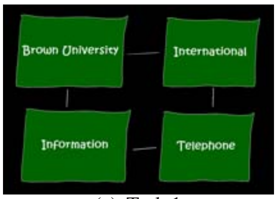
18 subjects from our University's general community participated in the study (8 male, 10 female). The average age was 25 with youngest 18 and oldest 42. Seventeen subjects use a computer very often and one somewhat often. 14 subjects occasionally sketch a diagram on paper and 12 subjects occasionally use a computer to create a diagram. About half the subjects sketch diagrams on paper and then transfer it to a computer. Fourteen subjects would choose a pen over a mouse for entering diagrams. Subjects were mostly neutral on whether their next computer should have a pen.

The physical setup consisted of a lightweight Hewlett Packard tc1100 with 1.2 GHz CPU and 512 MB RAM. The software used in the experiment was a custom application written for the study. The tablet was positioned in two configurations: for the Mouse conditions in an upright position on its base mount, and in a position the user decided was comfortable for the Pen conditions– most subjects rested it on the table top as in Figure 1. For Mouse conditions a standard sized USB keyboard and mouse were used instead of the scaled down tc1100 keyboard.

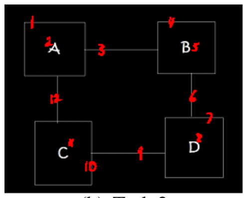
Figure 1: *The physical setup. Shown here is a pen technique and the subject has chosen to lay the Tablet PC on the table surface. (Out of view are the standard-sized keyboard and mouse used for non-pen techniques.)*

3.2. Experimental Task

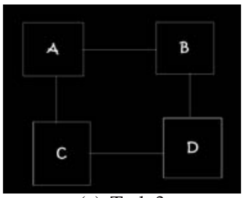
The task was to create a simple diagram consisting of four rectangles, text labels inside the rectangles, and connector lines between the rectangles. We designed six different diagramming tasks in an effort to isolate three different task variables, including: ordering of primitives, precision, and length of text. Although many other task variables could have been considered, we felt that the three chosen provided a reasonable coverage of the fundamental issues inherent to simple block and line diagramming. Each of the six tasks required creating four rectangles, text labels inside the rectangles, and connector lines between the rectangles. There were two sub-categories of tasks: copying from a physical sheet of paper (tasks 1 and 4) and tracing a diagram presented in an underlay (tasks 2, 3, 5, and 6). This attribute of the diagrams was meant to vary the level of precision subjects applied when creating them– it was expected the "copy task" would simulate more informal sketching and that the "trace task" would simulate more careful placement of primitives. For both of these sub-categories, we created tasks that involved both short (1 character) and long (about 12 characters) text strings, since pilot testing had shown attitudes toward text entry change significantly depending on text length. The long text was chosen from common words that were likely to be known by our University's community with international origins in order to simplify the task of typing or handwriting the words. Finally, for each of the trace tasks, we created two sub-tasks, one which allowed primitives to be entered in any order and one in which primitives had to be entered in a specific order. This latter condition was not intended to represent a real world task directly, but rather was intended to approximate non-linear creative thinking which obviates the optimized strategy of entering all similar primitives in batches.



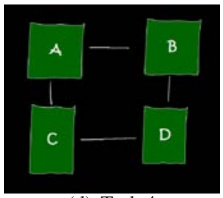
(a) Task 1



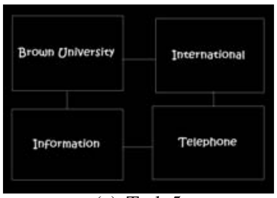
(b) Task 2



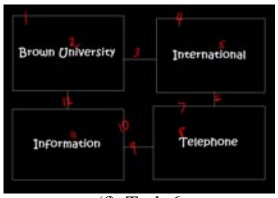
(c) Task 3



(d) Task 4



(e) Task 5



(f) Task 6

Figure 2: *Six tasks of varying text quantity, precision, and primitive entry order.*

Six User Interfaces. In an effort to cover as much as possible the range of different drawing user-interfaces, we identified three fundamental categories of drawing techniques, including: toolbar techniques that are the standard for self-disclosure and ease-of-use, literal drawing techniques that are arguably the most intuitive, and shortcut techniques that are potentially the most efficient. For each of these three categories, we implemented one technique for pen-input and an analogous technique for mouse and keyboard input. Table 1 describes the six techniques.

Mouse Toolbar most closely matches common diagramming user interface (see Figure 3). Mouse Shortcut offers a two-handed, keyboard modifier approach to selecting the primitive to enter. Mouse Drawing lets users literally draw the diagram, although the keyboard is used for entering text.

The three Pen techniques parallel the three Mouse ones with two exceptions. Text is hand drawn and the keyboard is not used. Pen Gesture does not use the common keyboard modifier but instead a drawn gesture (see Figure 4) to indicate if a line or rectangle should be created. The drawn

Table 1: *Description of the 6 UI techniques evaluated.*

Pen Drawing	Rectangles, lines, and text drawn with the pen are recognized as such.
Pen Gesture	Straight lines are recognized as lines. Lines with a "back trace" gesture are recognized as rectangles. Text is handwritten inside rectangles.
Pen Toolbar	The primitive mode is set on a toolbar. Text is handwritten inside rectangles.
Mouse Drawing	Rectangles and lines drawn with the mouse are recognized as such. Text is typed in at a cursor location inside a rectangle. The cursor can jump between rectangles by pressing TAB and SHIFT+TAB.
Mouse Shortcut	Mouse down through up events rubber bands a line. Holding the CONTROL key rubber bands a rectangle instead. Text is entered the same as for Mouse Drawing above.
Mouse Toolbar	The primitive mode is set on a toolbar. Text is entered the same as for Mouse Drawing above.

gesture is motivated by: 1) some pen systems do not have keyboards, 2) the gesture is efficient, 3) the gesture is easily recognized, 4) the gesture is not awkward, 5) the gesture does not look like text so can be used in a free-inking context, and 6) the gesture allows the simple transition from drawing to rubberbanding rectangles. Practical considerations like study duration prevented us from testing every combination, so we tried to select the most interesting parameters.

The recognition algorithm for the Pen and Mouse Drawing techniques is not general purpose, but instead was designed for robustness within the scope of the test diagrams. The algorithm simply tests if the last point of a stylus stroke was close to the first point; if the points are close then a rectangle is recognized, otherwise a line is recognized with the exception that any strokes drawn within a rectangle were considered text strokes (but no text recognition was actually done). The instructions given to subjects, however, were to "draw a line" to get a line, "draw a rectangle" to get a rectangle, and "print or cursively write text inside an existing rectangle neatly enough that a stranger could read it". Subjects were told to try to draw the shapes accurately, but not to worry if they were a little off.

3.3. Experimental Design and Procedure

We used a 6 x 6 within subjects factorial design where the independent variables were technique and task, and the dependent variables were completion time and time spent entering text. The techniques and tasks varied as described in section 3.2. The completion time is from when the subject pressed a "start" button until they pressed an "end" button

(a) (b)

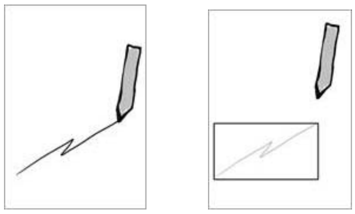
Figure 3: *a) The toolbar with 32x60 pixel buttons used in the study is on the left side of the application window. b) two rectangles and a line connecting them were created. The user typed the label "A" inside the left rectangle and then pressed the "TAB" key. Now the right rectangle has a yellow text input vertical line cursor indicating where typed text would appear.*

(after producing the diagram). "Time spent entering text" is defined as the time to type on the keyboard all labels for a diagram, or the time spent handwriting the labels.

The experiment began with the subjects completing a pre-questionnaire (age, gender, computer usage, experience using a pen device). Subjects were given a series of tasks to get familiar with using the pen on a Tablet PC. First, they watched a ~1 minute long "Finding a good position" video explaining how to draw on a Tablet PC (this video comes with Microsoft's Tablet PC). Second, to get a feel for drawing box-line-text diagrams, they spent ~5 minutes drawing a family tree or course schedule. Third, to get practice clicking on GUI buttons of various sizes, they searched for a movie review on a news site. Finally, to get a feel for free-form drawing, they spent ~2 minutes drawing an approximate map of the world without worrying about accuracy.

In the formal study, each subject used six techniques to perform six tasks for a total of 36 trials. We explained and demonstrated each technique and then subjects practiced making rectangles, lines, and text until they said they were ready to begin the six tasks. We used 6x6 Latin Squares to address ordering effects. After completing the six tasks with each technique, subjects answered a "post technique" questionnaire. Subjects were asked to react to seven statements using a 5-point Likert scale (1 = strongly disagree, 5 = strongly agree). The statements were: I found this technique was easy to learn; In general, I liked this technique for the tasks; In general, I found this technique easy to use; In general, I was distracted by this technique; In general, I was doing the tasks, not thinking about how to do them; In general, I liked entering text with this technique; and In general, I would be frustrated if I had to use this technique for these tasks.

After completing all trials, subjects completed a post-questionnaire which asked them to rate each technique using a 7-point Likert Scale (1 = strongly disliked, 7 = strongly liked) and commented on their experiences with them.



(a) (b)

Figure 4: *A rectangle is created with the Pen Gesture technique by first a) drawing a line that has some "back trace" within it and endpoints that will determine two opposite corners of the rectangle. b) shows the resulting rectangle when the pen is lifted. The grayed gesture remains to illustrate the relationship between the gesture and rectangle, but it disappears immediately in the application.*

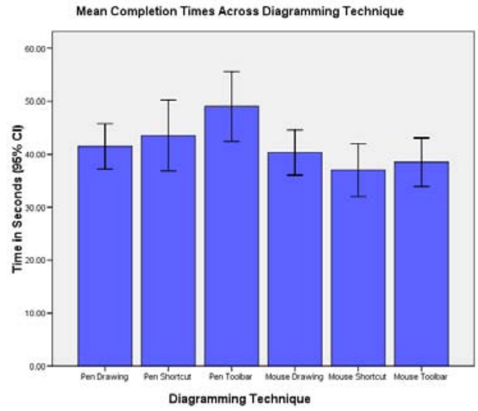


Figure 5: *Summary of task performance across techniques.*

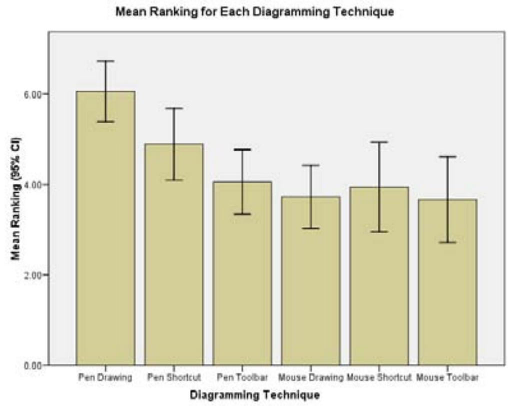


Figure 6: *Ranking of techniques by participants (1 = strongly disliked,7 = strongly liked).*

4. Results

Quantitative analysis. A repeated measures two-way analysis of variance (ANOVA) was performed with completion time as the dependent variable and technique and task as the independent variables. Note that when we analyzed the completion time data, we discovered that timings for eight (for five subjects) of the 648 (1.2%) trials were lost. To fill in the missing values, we used a statistical technique known as multiple imputation [Sch99]. This approach is a Monte Carlo technique in which the missing values are replaced by m>1 simulated versions, where m is typically small. In our case we used the procedure several times (with m=10) and ran significance tests on each one. In each case, the results were the same in terms of statistical significance when compared with an analysis using data from the 13 subjects with no missing values. Thus, we are confident our statistical tests are valid. The results of the two-way ANOVA revealed significant differences for technique ($F_{5,13} = 4.383$, $p < 0.05$) and task ($F_{5,13} = 181.94$, $p < 0.05$). Note we were unable to conduct a test for interaction effects between technique and task since we did not have enough subjects.

The overview of our data is shown in Figures 5, 6, and 7. A pairwise post-hoc analysis was conducted on the task completion times across the different techniques (with task collapsed) using Holm's Sequential Bonferroni correction method [Hol79] with 15 comparisons at $\alpha = 0.05$. Due to the Bonferonni adjustment, no technique was significantly faster or slower across all tasks, even if time to enter text was not considered. When separated into the 36 individual conditions (see Figure 7), tasks 2, 3, and 4 appear to be the fastest to perform and this makes intuitive sense as those tasks all had the short text attribute.

As one might expect, long and short text took (respectively) on average 26 and 4 seconds to write by pen, while long and short text took (respectively) 7 and 2 seconds to type. Also, while we did not measure text-entry times *including* transition time (i.e., time to physically move to and from widgets or between the mouse and keyboard), Figure 7 indicates the transition time even for our worst case condition (i.e., task 6 the long-text, ordered primitive condition) did not make the pen-based techniques faster than mouse-based ones.

Qualitative Analysis. The main results of the post-technique questionnaire are given in Figure 8. Subjects said Pen Drawing was easiest to use and least frustrating. Subjects also found all six techniques were easy to learn and they liked entering text with all techniques. However, they disagreed that Pen Drawing, Pen Gesture, and Mouse Gesture were distracting and felt neutral about Pen Toolbar, Mouse Drawing, and Mouse Toolbar. Subjects generally reported neutral or positive feelings about each technique– they were not negative about any of them.

The results of the post-study questionnaire (after all 36

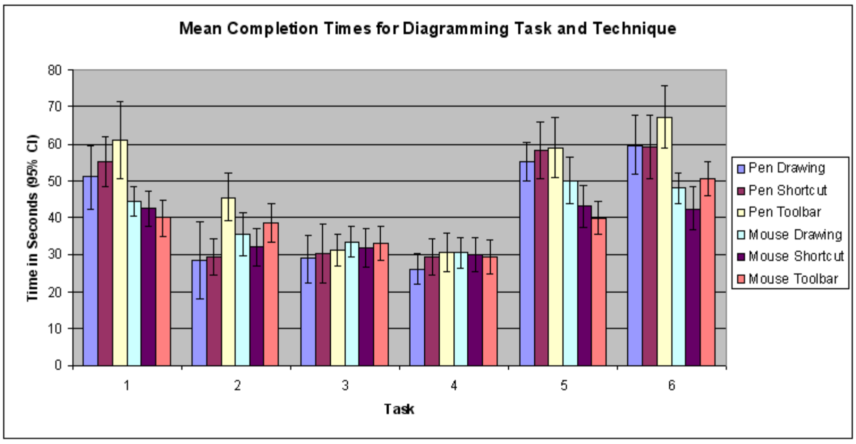


Figure 7: *Summary of technique performance across the 36 conditions.*

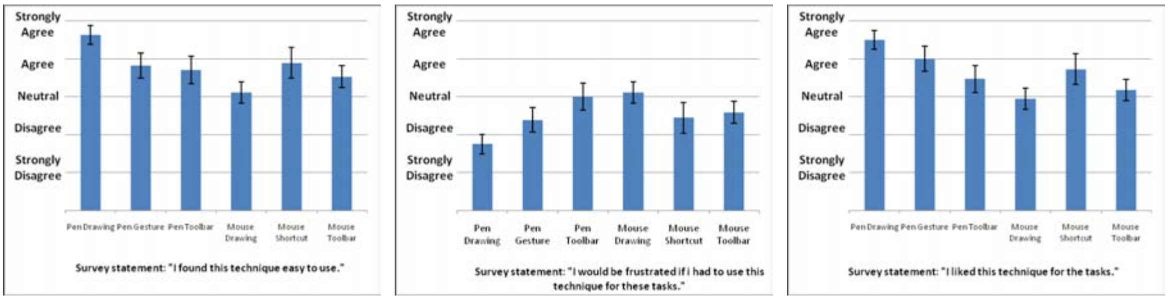


Figure 8: *Participants were asked the above survey question immediately after they used each technique (95% CI's shown).*

trials were run) included a ranking of the six techniques (see Figure 6). We conducted a Friedman test on each of these statements and found there were significant differences in subject responses ($\chi^2(5, N = 18) = 24.46$, $p < 0.05$). To further analyze this data, we ran a post-hoc analysis performing pairwise comparisons using the Wilcoxon Signed Rank test. We also used Holm's sequential Bonferroni adjustment [Hol79] with 15 comparisons at $\alpha = 0.05$. The results show that subjects rated Pen Drawing higher than Mouse Drawing (z = -3.499, $p < 0.0033$), Mouse Toolbar (z = -2.889, $p < 0.0038$), and Pen Toolbar (z = -3.46, $p < 0.00357$). These results indicate that although subjects did not perform the diagramming task faster or slower with any particular technique, they preferred Pen Drawing over Mouse Drawing, Mouse Toolbar, and Pen Toolbar.

The post-questionnaire also included open-answer responses to four questions. *Question 1: Did any of the interface techniques stand out as being clearly better than the others? If so which and why?* The number and percentage of subjects who gave each response follows the technique name in parentheses: Pen Drawing (11, 61%); Pen Gesture (5, 28%); both Pen Drawing and Pen Gesture (2, 11%); Mouse Toolbar (3, 17%); Mouse Shortcut (1, 6%); Mouse Toolbar (1, 6%). Interesting responses were:

- "The Pen Drawing technique was simple and easy to use. It seemed to go by a lot quicker."
- "[Pen Gesture] was the best because it offers the precision and speed of the shortcut commands (especially important for drawing rectangles), and it doesn't require you to 'switch modes'."

Question 2: Did any of the interface techniques stand out as being clearly worse than the others? If so, which and why? The results were: Mouse Drawing (7, 39%); Pen Gesture (4, 22%); Pen Toolbar (4, 22%); Mouse Drawing (2, 11%); Mouse Shortcut (0, 0%); and Pen Drawing (0, 0%). Interesting responses were:

- "I didn't like [Pen Gesture] so much. It took a few tries to get the hang of the squiggle, and I found I had to think more about what exactly I was doing."
- "The Pen Toolbar seemed pointless to me. Why have a toolbar for shapes you can just draw?"

Question 3: Do you feel that you would switch among all of the techniques depending on the task, or were there one or two techniques that you think you would use for most all *tasks?* The results were: would use keyboard-based text entry for large amounts of text (7, 39%); would use Pen Drawing exclusively (9, 50%); Pen Gesture exclusively (7, 39%). Interesting responses were:

- "Some were more pleasant (e.g., Pen Gesture) but I probably wouldn't use them in a deadline situation when command keys, etc. are much faster."
- "I would simply use the pen for everything. It's easier than going back and forth."
- "I would not use Pen Gesture unless I absolutely had to but I would probably alternate between Pen Drawing, Pen Toolbar, and Mouse Drawing and Mouse Toolbar."
- "No, I would not switch between the tasks because I would forget what technique I was using and then combine all of them when I should not."

Question 4: How did you hold the tablet while doing the techniques involving the pen? Was this comfortable for you? The results were: flat on the table was comfortable (10, 56%); flat on table initially, but that would become uncomfortable (3, 17%); in lap (2, 11%); like a pad of paper (1, 5%); and like a clipboard (1, 5%). Interesting responses:

- "It was very easy to use the pen and the tablet. I rested it on the table and wrote on it like it was a piece of paper."
- "For a few of them on my lap and for others I placed it on the desk, it was very comfortable."
- "At first I leaned the tablet on the edge of the table, but this turned out not to be ideal. Just flat on the table was best."

5. Discussion

We have two interpretations of this study's results: that it has and that it has not rejected the null hypothesis (i.e., that pen-centric input affords no significant advantage to keyboard and mouse input even for a 2D diagramming task).

It has not rejected the null hypothesis in that from a timed performance perspective the Mouse techniques were comparable to the Pen techniques (Figure 4). And even though subjects ranked the Pen Drawing technique significantly higher from a statistical point of view, that preference was not explored in depth here and it may be the case that under more realistic conditions (e.g., their job depended on it, long term use, richer tasks) that ranking may change.

This study has rejected the null hypothesis in that subjects did, in fact, rank the Pen Drawing technique higher than the other techniques despite the efficiency metric and that can not be discounted. A hypothesis for why Pen Drawing was preferred is that it is the most direct technique and we believe this is a concept worth additional investigation.

In either case, further studies are warranted. "Real-world scenarios" that evaluate head-to-head mouse and pen input could further support the hypothesis. There is a research barrier to this kind of study in terms of choosing or designing an interesting set of interfaces to use in the evaluation. We can imagine many possibilities to further investigate and test the "more direct input offers significant advantage over less direct input" hypothesis including:

- Develop a model that can predict a UI's level of directness
- Investigate more carefully the pen and mouse drawing techniques (which we argue are the two most direct methods) and the effect of absolute and relative input on level of directness.
- Investigate the relationship between task completion time and preference more deeply
- Re-run this study but vary the complexity of tasks and probe deeper into the ranking question.

Further studies may also reveal that no single hypothesis can summarize the user preferences under all scenarios but it varies depending on the specific task and the context in which this task is being performed.

The results indicate that somewhere between 1 and 13 characters (the average length of the long words used) is where performance might increase if pen-users transitioned to other input methods like keyboard or dictation. For example, the currently popular tablet form factor "clamshell" provides easy keyboard access at the cost of using a less convenient, angled writing surface. The study also suggests mouse-based drawing of primitives with recognized strokes may be worth studying further because subjects did not have the expected negative feelings towards Mouse Drawing (see Figure 8) and it shares the drawing aspect of the highest ranked technique (i.e., Pen Drawing). We hypothesize at some point as diagramming complexity increases subjects would dislike Mouse Drawing. While we did not see statistically significant differences in learnability and subjects agreed with the statement that all techniques were easy to learn, subject open-ended comments indicate Mouse Shortcut and Pen Drawing would benefit from better methods for teaching the gestures. Our study did not indicate the hardware itself was a serious problem, although we often hear comments that hardware available today does not "feel" like writing with pencil and paper.

While the interactions user performed were somewhat complex, the diagrams were simple. If more geometric information needs to be specified, we expect as did MacKenzie [MSB91] that the trends will continue and drawing and gestures, once learned, would excel for all metrics. Apet's more complex study suggests this may be valid [AK93].

When a decision had to be made such as what size buttons to use in the toolbar we tried to make choices most in favor of keyboard/mouse input. For example, we used toolbar buttons 60 pixels wide and 32 pixels high which are less common but easily targetable– many programs such as Microsoft Word and PowerPoint use only 22-pixel high toolbar buttons (e.g., for line and rectangle mode). Another example is subjects did not need to click on the text button, but could just start typing at any time. These design decisions which are not always applied in real-world apps would impact user satisfaction and performance and could be studied further.

No technique was always the worst performer. And it is worth noting that all six techniques could (and did in our prototype) co-exist in a single application. Given the post-questionnaire responses in section 4 there is variability in whether users feel they would want one or multiple techniques. This question may warrant further study.

This study presents evidence that metrics besides speed are important for desktop tasks and that the gold standard of mouse and keyboard UI's may not be liked better than a pen-based UI or hybrid environment. We found that even though mouse and keyboard interfaces were faster or comparable in speed to pen interfaces that subjects ranked pen interfaces higher than the mouse interfaces.

Another larger benefit to this work is we believe these user interfaces, nearly as-is, could be a sufficient front-end for creating usable, high-quality diagrams if coupled with an inference engine capable of matching high-level templates, such as the SmartArt used in Microsoft Office applications.

Future work should explore more complex diagramming and more general real-world computing tasks to learn how the trends reported might vary. Incorporating crossing-based pen techniques may be interesting to explore. Investigating how display area impacts subjects would be interesting– today's Tablet PC's display is relatively small, generally. Discovering and analyzing more gestures with useful qualities like the drawn gesture in 4 would be useful.

6. Conclusions

We presented a formal study intended to shed light on the potential benefit of pen-computing in desktop scenarios by challenging the null hypothesis that there is no significant advantage for pen-computing in desktop environments. Although our study does not conclusively reject this hypothesis, it does provide sufficient statistically significant evidence to warrant further investigation. Specifically, users preferred pen drawing over other techniques, even though the tasks were designed to allow optimized keyboard and mouse techniques and in fact users performed, if anything, faster with keyboard and mouse. Furthermore, even though the tasks evaluated were limited to drawing simple diagrams, we believe that the results are more widely applicable because the diagramming primitives used (rectangles, lines and text) can be thought of as representing building blocks for general application interactions, such as pointing, selection, and text entry. However, given that pen toolbar techniques were ranked equivalently to all the mouse techniques, we surmise that general adoption of pen input will be delayed until applications shift from their current mouse-optimized point-and-click metaphors to pen drawing-optimized mark-up metaphors.

Important insights from this work are: 1) in contrast to prior work we did not find pen-device to be twice as fast as the mouse-based device for specifying diagrams, 2) transition time between mouse and keyboard devices did not make pen faster for tasks/techniques, and 3) despite it being slower, subjects reported they liked doing the tasks with the pen.

7. Acknowledgements

We would like to thank Microsoft for supporting this work and the reviewers for helpful comments.

References

[AD04] ALVARADO C., DAVIS R.: Sketchread: a multi-domain sketch recognition engine. In *UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2004), ACM, pp. 23–32.

[AG04] APITZ G., GUIMBRETIÈRE F.: Crossy: a crossing-based drawing application. In *UIST '04: Proceedings of the 17th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2004), ACM, pp. 3–12.

[AK93] APTE A., KIMURA T. D.: A comparison study of the pen and the mouse in editing graphic diagrams. In *Symposium on Visual Languages* (1993), IEEE, pp. 352–357.

[Dav02] DAVIS R.: Sketch understanding in design: Overview of work at the mit ai lab. In *AAAI Spring Symposium* (2002).

[Hol79] HOLM S.: A simple sequentially rejective multiple test procedure. In *Scandinavian Journal of Statistics* (1979), vol. 6, pp. 60–65.

[LM01] LANDAY J. A., MYERS B. A.: Sketching interfaces: Toward more human interface design. *Computer 34*, 3 (2001), 56–64.

[LNPS05] LANE D. M., NAPIER H. A., PERES S. C., SANDOR A.: Hidden costs of graphical user interfaces: Failure to make the transition from menus and icon toolbars to keyboard shortcuts. In *International Journal of Human-Computer Interaction* (2005), vol. 18, pp. 133–144.

[MSB91] MACKENZIE I. S., SELLEN A., BUXTON W. A. S.: A comparison of input devices in element pointing and dragging tasks. In *CHI '91: Proceedings of the SIGCHI conference on Human factors in computing systems* (New York, NY, USA, 1991), ACM, pp. 161–166.

[Sch99] SCHAFER J. L.: Multiple imputation: A primer. In *Statistical Methods in Medical Research* (1999), vol. 8, pp. 3–15.

[SL03] SAUND E., LANK E.: Stylus input and editing without prior selection of mode. In *UIST '03: Proceedings of the 16th annual ACM symposium on User interface software and technology* (New York, NY, USA, 2003), ACM, pp. 213–216.

[SSD01] SEZGIN T. M., STAHOVICH T., DAVIS R.: Sketch based interfaces: early processing for sketch understanding. In *PUI '01: Proceedings of the 2001 workshop on Perceptive user interfaces* (New York, NY, USA, 2001), ACM, pp. 1–8.

[Tor06] TORRES D. A.: Evaluating a pen-based computer interface for novice older users. In *Assets '06: Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility* (New York, NY, USA, 2006), ACM, pp. 249–250.

[ZM06] ZELEZNIK R., MILLER T.: Fluid inking: augmenting the medium of free-form inking with gestures. In *GI '06: Proceedings of Graphics Interface 2006* (Toronto, Ont., Canada, Canada, 2006), Canadian Information Processing Society, pp. 155–162.