A Study Comparing Mouse-Only Input vs. Mouse-Plus-Voice Input for a Graphical Editor

Randy Pausch and James H. Leatherby

Computer Science Report No. TR-90-17
July 10, 1990

A Study Comparing Mouse-Only Input vs. Mouse-Plus-Voice Input for a Graphical Editor

Randy Pausch & James H. Leatherby

Computer Science Department
University of Virginia
Thornton Hall
Charlottesville, VA 22903-2442
(804) 982-2211
pausch@Virginia.edu

Abstract

Previous studies have measured the speed and accuracy of voice input versus devices such as keyboards and mice. Other studies have shown that users will naturally provide input via simultaneous channels. While these studies indicate that voice might be used to improve the efficiency of a human-computer interface, they provide little assistance in predicting how much difference would be made by adding voice as a parallel input channel for common tasks. This paper presents a user study which measures how long it takes users to create drawings with a graphical editor. Our control group used the mouse as an input device, and our experimental group used a discrete-word voice recognition system in combination with the mouse. The combination system reduced task completion time by as much as fifty-six percent, with an overall reduction of more than twenty-one percent. We used an existing graphical editor originally designed to receive mouse input; we expect that a system explicitly designed to use parallel input channels could significantly improve our results.

This work was supported in part by the Virginia Engineering Foundation.

Introduction

Various empirical studies have measured the speed and accuracy of voice input versus devices such as keyboards and mice [1-5]. Other studies have shown that users find it natural to simultaneously manipulate separate input devices with each hand [6] and that users will naturally provide simultaneous voice and gesture input [7, 8]. Although these studies indicate that voice could be used in combination with other devices, they do not provide any insight into how effective this might be for common tasks. We have recently performed a study which measures the effectiveness of combining voice input with a mouse for creating monochrome graphic artwork.

Description of the Study

The effectiveness of any system depends heavily on the task to be performed. For any input device, or combination of devices, one can choose a task for which the device(s) are best. We chose to measure the common, well understood task of creating monochrome "line art" with an interactive graphical editor. We used the MacDraw program (Claris, version 1.9.6), running on a monochrome Macintosh computer. In order to choose a representative set of tasks, we randomly selected figures from three popular technical journals: *Communications of the Association for Computing Machinery, Science,* and the *Journal of the American Institute of Chemical Engineers.* Eight total drawings were used in the experiment; they are presented (smaller than actual size) in Appendix A.

An experimental group of sixteen subjects was used; all were either graduate or undergraduate students at the University of Virginia. Most subjects had previously used a mouse, but none were expert MacDraw users. The subjects were randomly divided into two groups; the control group used the mouse-only version of the system and the experimental group used the mouse-with-voice version. Each subject first created a practice drawing and was then timed while creating four drawings. For each drawing, the subject started with a blank MacDraw screen and a printed copy of the artwork. The subject was allowed to study the artwork as long as desired before beginning the timed task. Both the control and experimental groups used the keyboard to type text for the MacDraw `text` command, but were not allowed to use the keyboard to invoke any other MacDraw commands.

Voice input was provided by an Articulate Systems Voice Navigator XA™, providing speaker-dependent, isolated word recognition. Each subject wore a headset with a back-cancelling microphone and worked in a quiet environment. Voice training required approximately twenty minutes per subject. Our vocabulary contained nineteen words, shown in Table 1, all of which were also available via either static menus on the left of the screen or pull down menus on the top of the screen. If a command was not available by voice, or if the subject forgot its name, the subject was forced to access that command via the menus.

Arc	Constrained Line	Duplicate	Line Segment	Polygon	Select All
Arrange	Copy	First Level	Oval	Rectangle	Text
Arrow	Cut	Freehand	Paste	Rounded Rectangle	Undo
Clear					

Table 1: Vocabulary Used in the Experiment

Results

As shown in Table 2, subjects created each of the drawings more quickly when using voice in parallel with the mouse. The reduction ranged from 8.79 to 56.78 percent and depended on the contents of each picture.

The *average speedup per picture* was 25.17 percent. This calculation, however, ignores the fact that the individual pictures had a large variation in their complexity; by counting each picture's speedup equally in the average, we bias the result towards the simpler pictures. For example, a picture whose drawing time decreased from 20 seconds to 10 seconds would have a 50 percent reduction, and a picture whose drawing time decreased from 1000 seconds to 900 seconds would have a 10 percent reduction. Computing a 30 percent average reduction for these two drawings is technically correct, but a better measure of time reduction is obtained by dividing the sum of the total raw times. In this example, dividing 910 by 1020 yields 89.2, or a 10.8 percent overall reduction in task time. When we perform this calculation, we find an *overall time reduction* of 21.23 percent in our study.

Drawing Number	Mouse Only	Mouse Plus Voice	Percent Speedup
1	956	872	8.79
2	212	194	8.49
3	236	102	56.78
4	407	260	36.12
5	339	283	16.52
6	306	250	18.30
7	567	441	22.22
8	161	106	34.16
average speedup per picture			25.17
total	3184	2508	21.23

Table 2: Average Task Times (all times given in seconds)

Because a relatively small number of subjects was used, we wanted to ensure that our random selection had not assigned more talented subjects to the experimental group. We had our subjects switch input groups and redraw the pictures. Both groups improved due to practice, but the voice plus mouse group was again better, indicating that system differences, not subject differences, account for our results.

The voice recognition system failed to correctly recognize four percent of the utterances; roughly three percent were ignored, and one percent were misinterpreted. The nature of the task made it possible for these errors to be quickly corrected (most MacDraw commands have an inverse, and there is also a general purpose undo command). None of our subjects found the error rate to be significant.

Discussion

We believe that using voice in parallel with the mouse provides substantial speedup for two major reasons. First, the user is no longer required to make large mouse motions to reach menus. Second, the task has many operations where commands are given in conjunction with screen locations, which are naturally parallel operations.

Although other techniques can be used to avoid menu travel time, each has drawbacks not present with voice. Keyboard accelerators, sometimes called *hot keys*, allow the user to invoke menu functions with a keyboard character. This allows parallel input via two hands, but requires that users memorize the keyboard mappings. Users who use a system regularly, but not frequently, are often able to remember command names but not their keyboard accelerators. Even for daily users of a system, memorizing a large number of keyboard commands is a burden.

Menus which *pop up* or can be positioned near the user's work area (sometimes called *tear away* or *push pin* menus), also reduce travel time. They have two drawbacks: they obscure some portion of the work space,

and they force the user to perform a low-level context switch for how the mouse is used. Voice provides the advantages of parallel input without requiring user memorization or consuming screen space. Menus continue to provide the advantage that all possible choices are displayed for those occasions when the user cannot remember the names of commands. A final note about menus concerns hierarchy. For systems with a large number of commands, menus often become hierarchical, and users must remember both the name of a command and where it lives in the menu hierarchy. Users must often search through the menu structure to find a command, even though they already know the command's name. This has implications for system design; we note that our voice recognition software provided a default voice command template for the MacDraw commands, but forced the user to *voice navigate* the menu hierarchy by saying the name of the MacDraw `Edit` pull down menu before allowing the user to say `Copy`. We found this approach awkward and instead created a single level of voice commands which were available at all times.

We originally expected that we would need to provide distinctive command names for the benefit of the speech recognition hardware. Instead, we were able to use the names given in the MacDraw documentation for all but one command, `Line`, which was frequently misrecognized in pilot trials. We changed the name of the `Line` command to `Line Segment` and the resulting error rate was below four percent. (We also rationalized the change by observing that our name was more mathematically correct...) One final observation was that many of our subjects who used voice input did so over two sessions which were four weeks apart. We found these subjects were able to use their previous training templates with little or no difficulty; we began their second session by asking them to say each of the nineteen words once, and we had to retrain an average of one word per subject before beginning the second session.

Conclusions

While previous research has shown that users will naturally provide voice input in parallel with other devices, most previous studies on voice input have been *versus* studies, where voice is compared against some other input device. Our study measured the reduction in task time when existing, low cost voice input technology was used in *conjunction* with the mouse. Our task domain was the creation of monochrome line art, and our subjects demonstrated an overall time reduction of over twenty-one percent. The drawing package was not originally designed to use voice in combination with mouse input and we strongly suspect that our results could be improved by altering the underlying software. In that sense, we view twenty-one percent as a *lower bound* for the reduction of task time when adding voice to graphical editors.

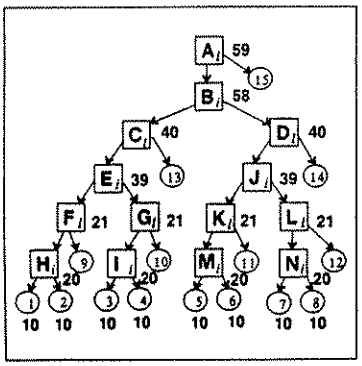
Our next user studies will measure parallel voice and mouse input for other tasks, and will examine issues such as display size, vocabulary size, and the simultaneous use of voice with two-handed input.

References

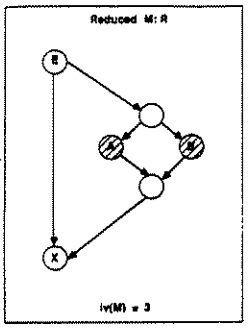
1. A. Biermann, R. Rodman, D. Rubin and J. Heidlage, Natural Language with Discrete Speech as a Mode for Human-to-Machine Communication,, *Communications of the Association for Computing Machinery 28*,28 (June, 1985), 628-636.
2. J. Leggett and G. Williams, An Empirical Investigation of Voice as an Input Modality for Computer Programming, *International Journal of Man-Machine Studies 21*(1984), 493-520.
3. I. Benbasat, A. S. Dexter and P. S. Masulis, An Experimental Study of the Human/Computer Interface, *Communications of the Association for Computing Machinery* 24,11 (November 1981), 752-762.
4. G. K. Poock, Voice Recognition Boosts Command Terminal Throughput, *Speech Technology 1*, 36-39.
5. J. M. Nye, Human Factors Analysis of Speech Recognition Systems, *Speech Technology 1*, 50-57.
6. W. Buxton and B. Myers, A Study in Two-Handed Input, *Human Factors in Computing Systems*, April 13-17, 1986, 321-326.
7. R. Bolt, Put-That-There: Voice & Gesture at the Graphics Interface, *Computer Graphics, 14*,3 (1980), 262-270.
8. A. Hauptmann, Speech and Gestures for Graphic Image Manipulation, *Human Factors in Computing Systems (SIGCHI)*, 1989, 241-245.

Appendix A:

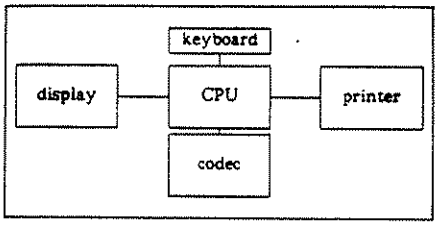
Drawing #1



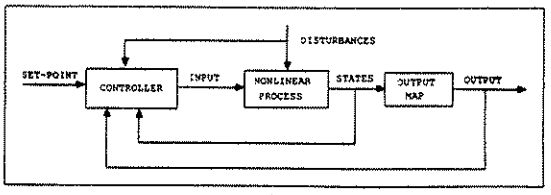
Drawing #2



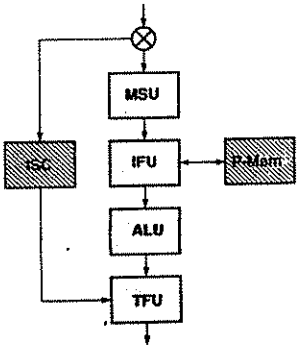
Drawing #3



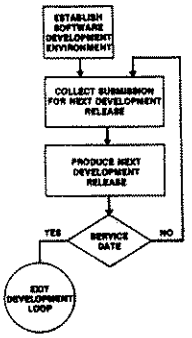
Drawing #4



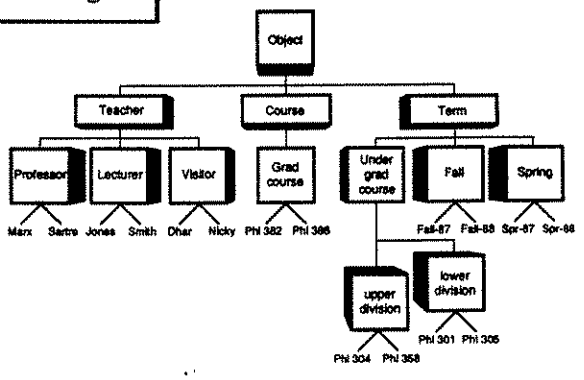
Drawing #5



Drawing #6



Drawing #7



Drawing #8