Buxton, W. (1994). Two-handed document navigation. *XEROX Disclosure Journal*, 19(2), March/April 1994, 103-108

TWO-HANDED DOCUMENT NAVIGATION

William A. S. Buxton

Proposed Classification
U.S. CL. 340/706
Intl. CL. G09g 3/02

Many current computer systems have a pointing device, Such as a “mouse,” for use with a direct manipulation graphical user interface (GUI) for manipulating information in Screen workspace regions called “windows". A characteristic of direct manipulation GUIs is the display and manipulation of graphical display objects (variously called “icons”, “tools” and “widgets) in order to interact with and accomplish tasks on the computer system. Graphical display objects perform two roles: they function as a *display* which conveys information about their function, or about an object on which a task is being performed, and as a *control* that the user manipulates to accomplish a function. Different graphical display objects are displayed for different functions. From the perspective of the display role of the GUI, the GUI can be said to be *space multiplexed.* However, virtually all interaction with graphical display objects is undertaken using a single input device (e.g., the mouse.) Hence from the control perspective, the GUI can be said to be *time multiplexed.*

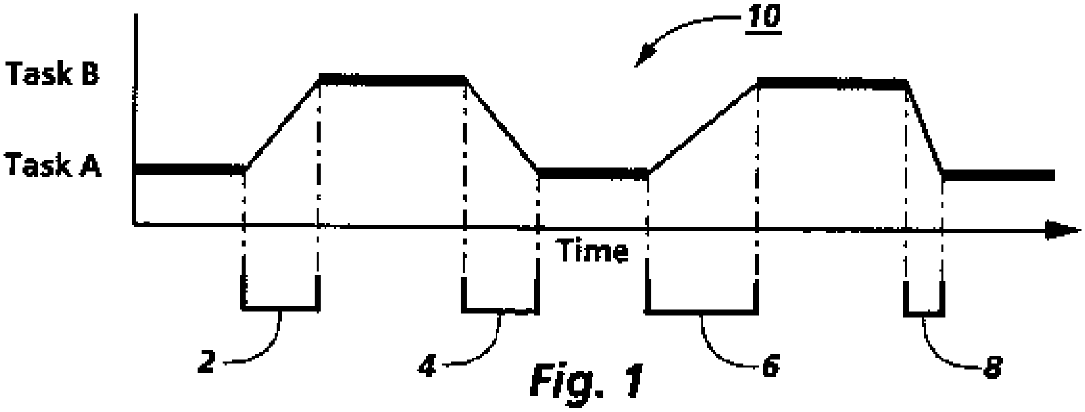

Fig. 1

Time multiplexing constrains the GUI to be serial in nature, since the user typically cannot perform two functions simultaneously using only one input device. More importantly, a typical sequence of actions includes an action for *selecting* the display object before actually performing the function, and the process of selection takes a quantity of time which maybe referred to as an *acquisition time.* In addition to taking time, a "select display object" action may disrupt the flow of the users work progress or train of thought. The graph 10 in Figure 1 illustrates acquisition times involved in performing a series of alternating tasks A and B in a time multiplexed GUI. The time intervals denoted by reference numerals 2, 4, 6 and 8, represented by the sloping lines in the graph, are the acquisition times needed for the user to alternately select the graphical display objects that represent tasks A and B.

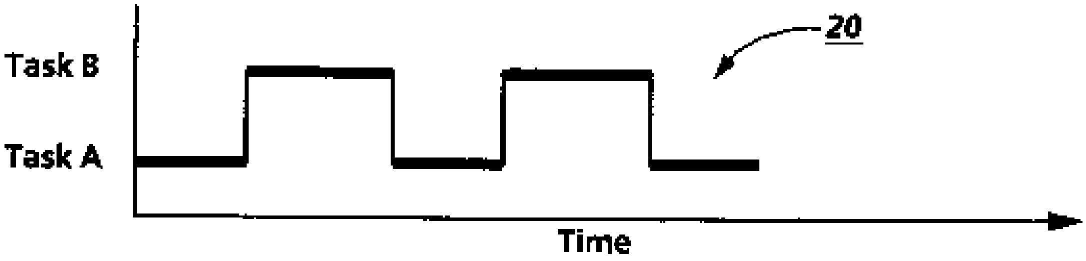

Fig. 2

The technique described here proposes to significantly reduce or eliminate acquisition time by assuming a distinct input device to each of a user's two hands, thereby controlling selection of the graphical display object for, and the performance of, each task with different input device. The use of two hands in a direct manipulation GUI is an extension of the use of two hands in many everyday tasks, where a person's nondominant hand works in a complimentary fashion with the dominant hand to perform tasks ranging from simple to complex. The graph 20 in Figure 2 illustrates that the acquisition times involved in performing a series of alternating tasks A and B is eliminated with the addition of a second input device. This is largely because each task has its own controller (i.e., input device), and each hand is able to return to a "home" position after selecting and performing a task, in preparation for selecting and performing the next iteration of the task. Therefore, task switching is reduced or eliminated.

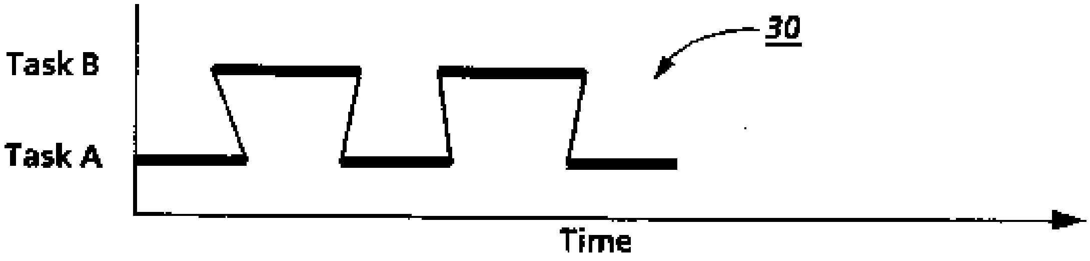

Fig. 3

Having independent simultaneous control over each task affords the ability to perform the two tasks together. For example, a user may drag an object using the input device assigned to the dominant hand while scrolling to a desired location in a document with the input device assigned to file nondominant hand. Or, in the context of graphical drawing application, the user may simultaneously scale and position an o14ect in a scene[1]. The graph 30 in Figure 3 illustrates that simultaneous action may actually lead to *overlapping* task times in performing alternating tasks A and B.

Of course, real Systems involve switching among more than two tasks. However, adding additional input devices does not accomplish a further increase in time savings or productivity. Rather, the direct manipulation technique proposed here is premised on the thesis that all direct manipulation actions can be categorized in terms of only two tasks: a *foreground* task and a *background* (auxiliary) ask. Foreground tasks are performed with the dominant hand, and background tasks with the nondominant hand. By categorizing tasks this way, and by providing a transducer suitable for each task, significant improvements in directness and manipulation can be achieved.

The two-handed direct manipulation technique can be applied generally to the task of navigation within a window, since navigation is a prime example of a background task. In particular, a specific application of this technique will be described in the context of navigating, or scrolling, through a document. In a non-computer context, when proofreading or annotating a document, one writes on thee document with the dominant hand while the nondominant hand typically turns the pages and keeps track of relevant passages or pages, having the ability to turn pages at an even rate to browse a document, or to turn a single page or many pages at once, as when jumping to a particular page.

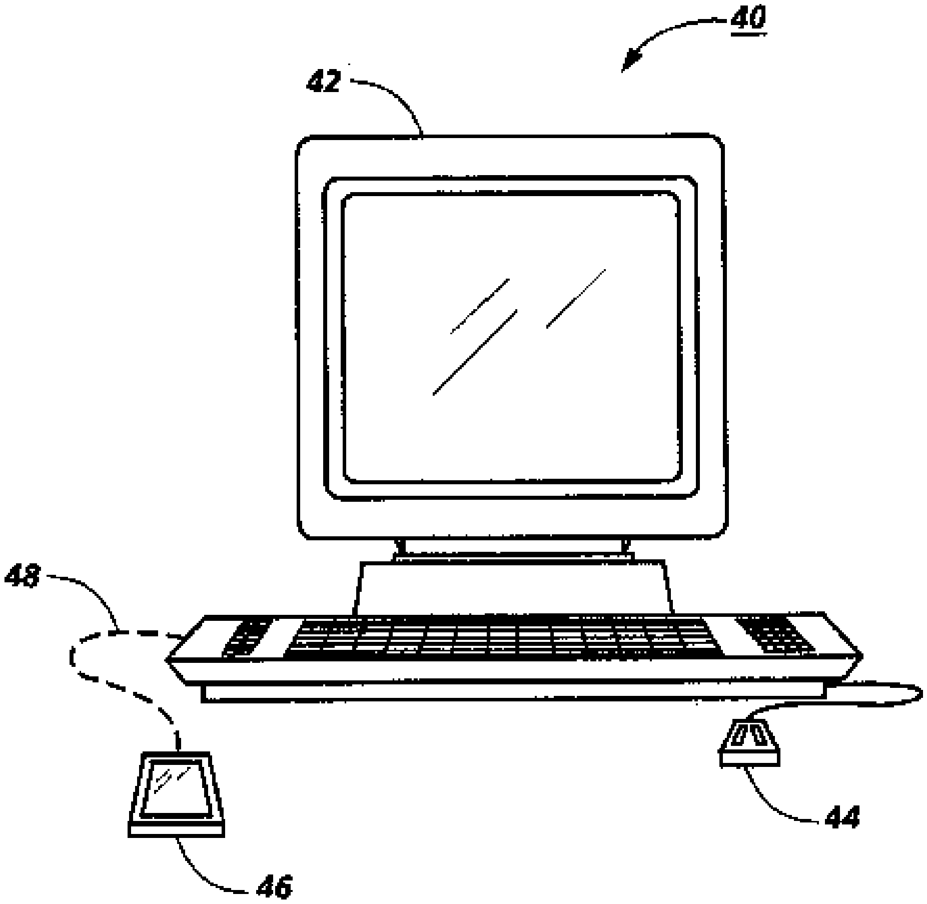

Fig. 4

In the context of a computer system, multi-page documents typically cannot be displayed in complete form in the limited display space available on most existing computer screens. Therefore, graphical display objects such as scroll bars are provided for navigating through the displayed document to reach portions not currently displayed. Scroll bars, like other graphical display objects, perform two roles: they function as a *display* which shows the user's current location in a document, and as a control with which tile user can change location within the document. Each window requires at least one and possibly two such scroll bars (one for each document dimension.) A user in a typical computer system must interrupt a task being performed, say, for example, editing the document, in order to access the scroll bar and bring the next portion of the document into view. This requires acquisition time, and slows down the user's progress. Moreover, scroll bars are typically designed to occupy a small space, and are typically very narrow. Because of their narrow size, a scroll bar display object, which is manipulated through the use of a mouse, is a very difficult target to acquire and, in fact, it has been shown that the difficulty of target acquisition is constrained by the shorter of the height and width of the scroll bar and that its acquisition is prone to error. [2]

The two-handed direct manipulation technique for document navigation proposed here uses a touch sensitive surface, such as a small touch tablet, as the input device for the nondominant hand and uses several simple gestures to accomplish document navigation. Figure 4 illustrates a computer system 40 configured to make use of the present technique with a display 42 and two input devices 44 and 46; mouse 44 is used for input by the dominant hand, and touch surface 46 is used for input by the nondominant hand. The two devices are shown in a configuration for a right-handed person. The surface of input device 46 can sense when it is touched and senses the location of the touch. This information is communicated to the computer and used to navigate within the document. Suitable devices for touch surface 46 include a touch tablet of any suitable size, including sizes as small as 2 or 3 inches square. Touch surface 46 may be directly connected to computer system 40 by way of a physical connection 48, or touch surface 46 may communicate with a receiver (not shown) in computer system 40 by way of an infrared or other wireless communication method. A small wireless touch tablet developed at Xerox PARC and suitable for use in the present application as input device 46, is called the “PARCtab.” [3] Contact with the touch surface 46 maybe made by touching the surface with a finger or using a stylus device.

The mapping of the input gestures to the document navigation commands summarized in Table 1 below, matches the functionality of existing scroll bars and will work with both linear and two dimensional navigation.

TABLE 1

Function	Gesture	Operation
Smooth scrolling	touch-slide-release	sliding the finger over the touch surface causes the document to scroll synchronously in the direction of tile motion (e.g., up or down, side to side), at a speed directly related to the speed of the motion, and by an amount determined by the amount of the motion; this gesture mimics pulling the document with finger;
Page turning	touch-rapid flick-release	a single rapid flicking motion over the touch surface causes the document to move in the direction of the flick by one page; this gesture mimics lifting/turning the page of a book;
Jumping to location	Touch-release	touching the touch surface in a position that is relative to a page location within the document, for example, the top of the touch surface could correspond to the beginning of the document.

The gestures have been selected to be both intuitive and easily distinguishable from one another. The reference to movement between and to “pages” in a document may correspond to either the actual physical pages of the document when the document is in tangible form, or to portions of the document that are previous or subsequent to the portion currently in the window.

REFERENCES:

[1] Buxton, W. & Myers, B. (1986). A Study in Two-Handed Input. *Proceedings of CHI' 86,* 321-326.

[2] MacKenzie, I.S. & Buxton, W. (1992). Extending Fitts' law to two-dimensional tasks. *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI’92)*, 219-226.

[3] Weiser, M. (1991). The computer for the 21st century. *Scientific American,* September 1991, 66-75.