

**METHOD AND SYSTEM FOR EMULATING A  
MOUSE ON A MULTI-TOUCH SENSITIVE  
SURFACE**

FIELD OF THE INVENTION

[0001] This invention relates generally to touch-sensitive display surfaces, and more particularly to emulating a mouse by touching a multi-touch sensitive display surface.

BACKGROUND OF THE INVENTION

[0002] With personal computers, there are two basic ways to control the movement of a cursor on a display screen: indirect and direct. In the most common way, a mouse or a finger on a touch pad is moved on a horizontal work surface, such as a tabletop, desktop or laptop, while the cursor moves on a vertical display surface. The input and display spaces are disjoint. With touch-sensitive direct-touch display surfaces, the cursor follows the movement of a finger or stylus in direct contact with the display surface, and is usually positioned directly under the contact point. The display space and the input space are the same space and are calibrated to coincide.

[0003] In cursor control, two modes are typically recognized for manipulating the cursor: positioning and engagement. Positioning mode simply moves the cursor over the displayed content without explicitly altering or actively interacting with the content, while engagement actively interacts with the content, e.g., moving a selected window or changing the appearance of the selected content. In a traditional desktop environment, positioning the cursor is typically done by moving the mouse; engagement is achieved by pressing one or more mouse buttons and possibly also moving the mouse. Typical operations in the engagement mode include dragging, i.e., moving the cursor with a mouse button depressed, and clicking and double-clicking, i.e., quickly pressing and releasing a mouse button once or multiple times.

[0004] Note that typically, while positioning may cause visual changes in the displayed contents, the changes are incidental to the movement of the cursor; the changes are temporary, provided by the system/application, and are intended as feedback for the user. For example, some graphical user interface (GUI) elements provide 'ToolTips' that are triggered by a mouse-over; when the cursor is placed over such an element, an information bubble is displayed. As another example, when the cursor is moved into and out of a GUI element, the element may change its visual appearance, e.g., highlighting and un-highlighting itself to indicate that it is an active element. It is not until or unless a mouse button is activated that engagement occurs.

[0005] One of the more fundamental challenges for direct-touch input is that users may wish to move a cursor across a touch-sensitive display without engaging any 'mouse' buttons, e.g., simply move the cursor over an icon. However, when a user touches a touch-sensitive surface, it is difficult for the system to detect whether the touch was intended to simply move the cursor or to interact with content, e.g., to 'drag' content with the cursor, as is done with indirect-control by holding down the left mouse button during the movement.

[0006] Thus, direct touch systems suffer from a different variant of the well known 'Midas touch' problem, i.e., every

touch is significant, see Hansen, J., Andersen, A., and Roed, P., "Eye gaze control of multimedia systems," ACM Symposium on Eye Tracking Research & Applications, 1995.

[0007] It is instructive to consider how other touch surfaces deal with this problem, even though most are not designed for a large touch-sensitive display surfaces.

[0008] The touch pad found on most laptop computers usually also includes left and right mouse buttons. There is also a mechanism to switch between modes without using the buttons. A user can switch between moving the cursor and dragging the cursor by tapping once on the pad, and then quickly pressing down continuously on the pad to drag the cursor. This sequence is recognized as being similar to holding down the left mouse button with indirect-control.

[0009] A second problem on a touch-sensitive display surface is that it can be difficult to precisely position a cursor with a relatively 'large' fingertip because the finger can obscure the very exact portion of the display surface with which the user desires to interact.

[0010] This problem can be solved by offsetting the cursor from the touch location. However, this forfeits one of the big advantages of a direct input surface, that is, the ability to directly touch the displayed content to be controlled.

[0011] Some resistive or pressure-based touch-sensitive surfaces typically use the average of two consecutive finger touch locations as the displayed position of the cursor. Laptop touch pads provide a single point of input. However, these are indirect input devices, and they do not address the problems of fluidly switching between positioning and engagement mouse modes. In the case of a laptop touchpad, auxiliary buttons may be provided to address the issue of fluidly switching between modes, but this does not solve the problem of having to rely on additional indirect input devices.

[0012] U.S. patent application Ser. No. 11/048,264, "Gestures for touch sensitive input devices," filed by Hotelling et al. on Jan. 31, 2005, describes methods and systems for processing touch inputs for hand held devices from a single user. That system reads data from a multipoint sensing device such as a multipoint touch screen. The data pertain to touch input with respect to the multipoint sensing device and the data identify multipoint gestures. In particular, the systems described are typically held in one hand, while operated by the other hand. That system cannot identify and distinguish multiple touches by different users. That is, the system cannot determine if the person touching the screen is the same person holding the device or some other person. Because the device is hand held, the number of different gestures is severely limited.

[0013] One direct touch-sensitive surface U.S. Pat. No. 6,670,561, "Coordinates input method," issued to Aoki on Dec. 30, 2003 uses an average of two consecutive touch locations as the position of the cursor. However, with this particular technology it is not possible to detect whether one or multiple locations were simultaneously touched, which limits the usefulness of the device. For example, the device requires a dedicated on-screen 'right click mode' button to specify whether touches should be interpreted as left clicks or right clicks. This solution does not support positioning mode at all, avoiding the issue of how to emulate moving the cursor without holding down a button.