DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0020] I describe improvements to graphical user interface systems, and in particular to methods and apparatuses for implementing a touch-sensitive graphical user interface. According to an aspect of my invention, a touch-sensitive input overlay is presented on the graphical user interface in response to a touch input on a display device, which includes a touch input device, such a series of capacitive or resistive sensors disposed over the display device. The touch-sensitive input overlay allows a user of a computer system to perform entry options without the aid of a traditional keyboard or mouse, but rather by touching one or more entry options presented on the touch-sensitive input overlay. The GUI system described herein is dynamic and flexible—allowing presentation of a number of unique touch-sensitive input overlays in variable locations on the display device. These and other advantages of the invention will be apparent to one of skill in the art upon review of the accompanying figures and the detailed description below.

[0021] Turning first to **FIG. 1**, it illustrates a touch-sensitive input overlay **16**, which is disposed over a traditional GUJI **6** presented on a display device **4**. According to an aspect of the invention, the display device **4** includes a touch input device that is responsive to physical contact. Such devices are commercially available and generally known in the art.

[0022] The GUI **6** comprises a series of visual indicators, which include control boxes **8**, **10**, and **12** for data entry, but can also include navigation entry fields (not shown) such as a tree-hierarchy. For instance, the control **12** is shown with a push-down button **14**, which opens to a list box **30**. Other types of controls can include combo boxes, push and toggle buttons, progress indicators, scroll bars, window edges (that facilitate resizing of a window), and other devices for display, inviting, responding, or accepting information between a user and a computer program.

[0023] I note that a "control" is an area or entry dialog/window into which data can be entered (note that a "control" is sometimes called a "widget" in Unix environments). A GUI comprises a plurality of controls, and in a normal GUI environment, a keyboard and mouse are shared among a number of controls. However, when the keyboard or mouse "focuses" on a particular control, that control has the attribute of receiving the keyboard or mouse entries (e.g., from a message queue of a thread that created it). Thus, as a particular control is selected, it "has focus" in the GUI. There can be only one focus (i.e. control) active in the GUI at any given time.

[0024] The GUI **6** does not have to be a specially programmed GUI—that is, it does not have to be a GUI programmed for touch-sensitive input. And herein lies an advantage of my invention: using an off-the-shelf touch input device, such as a touch-sensitive display, and the methods and techniques described herein, a highly flexible and useful touch-sensitive input overlay for the GUI is possible that either replaces or complements traditional data entry and navigation tools used with a standard GUI.

[0025] When a user touches within a pre-set area around a control (e.g., **8**, **10**, **12**) (hereinafter referred to as a "parent control") on the GUI **6**, the touch-sensitive input overlay **16** appears on the screen **4** in the proximity of the parent control (now having focus), effectively overriding standard processing of data/control selection and entry. A parent control delineator **13** is shown that correlates the parent control **12** to the touch-sensitive input overlay **16**.

[0026] According to one embodiment, the touch-sensitive input overlay is animated onto the screen from one or more points corresponding to the parent control to multiple points corresponding to the ultimate location touch-sensitive input overlay. For instance, rather than simply appearing in its fully rendered state, the touch-sensitive input overlay is gradually expanded or "faded-in" from the parent control to its full-size adjacent to the parent control—not so fast that it cannot be detected by a user's eye, but not so slow that it consumes too much time.

[0027] In another embodiment, the touch-sensitive input overlay can fade-in or pop-up on the screen and a portion of the border of the touch-sensitive input overlay closest to the parent control is delineated in a position corresponding to the parent control so as to identify the touch-sensitive input overlay with the parent control. For instance, the border can be partially removed, highlighted, or a line drawn to the parent control from a point along the border of the touch-sensitive input overlay. In yet another embodiment, the background of the graphical user interface can be faded out or turned into non-active color schemes (the standard windowing technique for highlighting the active dialog window), while the touch-sensitive input overlay is highlighted or turned into the active color schemes.

[0028] The objective in each of these techniques is to aid in allowing a user to identify the parent control for the touch-sensitive input overlay.

[0029] The touch-sensitive input overlay **16** can have a number of embodiments, which are pre-selected to best match the individual control parameters—such as control purpose (data entry or navigation) and type of data to be entered (numerical, list box, computed, user prompted, etc.). According to one embodiment, the position of the touch-sensitive input overlay **16** relative to the parent control depends on the parent control's location within the display and the unused GUI area within the proximity of the parent control. It is desirable to place the input overlay in a position where it is least obtrusive to adjacent controls or other on-screen information.

[0030] As shown in **FIG. 1**, the touch-sensitive input overlay **16**, which is suited for navigation in a list box, includes a number of touch entry options including a plurality of navigation arrows **18** and **20**, a "reset" option **22** (for resetting a control), and an "enter" option **24** (for entry of a selected data item in a list box **30**). Optionally, a "move" option **26** (for allowing a user to move the input overlay **16** to a different location on the display **4**), and a "close" option **28**, for exiting the touch-sensitive input overlay **16** (i.e., making it disappear), can be included. However, it is often preferred to minimize not only the complexity of the touch-sensitive input overlay **16**, but also the real estate. Thus, in another embodiment, only strictly essential options are include in the touch-sensitive input overlay **16**. Tasks such as closing the touch-sensitive input overlay, for instance, can be achieved simply by selecting another control and giving it focus.

[0031] A touch tool **32**, is also shown, which can be a plastic pointer or, preferably, a human finger. The touch tool