[0044] In the example method of transitioning shown in FIGS. **6-8**, the process of transitioning between a high-resolution input mode UI and a low-resolution input mode UI is directed to achieve a particular visual effect. In general, the visual effect can be understood by envisioning that the user is viewing the screen through a pinhole. As the display screen is moved towards the user, which would be the typical motion when the user wants to transition from a mouse-based input to a touch-based input, the edges of the pinhole are obscuring the outer portions of the display screen. In other words, items at the outer edge of the screen appear to be moving outward and off the edge of the screen, or being pushed or slid off, so that the view is essentially zooming-in to the center portion of the screen. Once the user is not able to see the menu bar, the dock, the scrollbar, etc., it is likely that the user's natural inclination is to not want to use those items anymore. However, the transition may appear smoother, for example, because the entire view is zoomed-in by the same percentage, and may result in a low-resolution input mode UI that is more similar to the high-resolution input mode UI (and vice versa). Therefore, the transition may be less jarring to the user, and thus, might seem more natural. A more natural, subtle transition might be less disruptive to the user's workflow. In addition, the percentage of zoom applied by transition effect **700** may be customizable by the user.

[0045] Even though certain items might be obscured by transition effect **700**, for example, because they appear to have slid off the screen, a view of these items can be regained (the items are retrievable). FIGS. **9**A and **9**B show an example method of un-obscuring items that have been slid off the screen according to embodiments of the invention. FIG. **9**A shows low-resolution input mode UI **800** displayed on screen **601**. If the user would like to view items that have been slid off the screen due to transition effect **700**, the user can use a touch input, for example, the user can place his or her hand (or one or more fingers) on the screen and move the hand or fingers in a direction, to un-obscure the items. As shown in FIG. **9**A, for example, the user wishes to view menu bar **603**, so the user places his or her hand on screen **601** and moved to hand downward to "pull" the menu bar back onto the screen. FIG. **9**B shows the result of the pulldown. The items of UI **800**, including application window **811**, thumbnail images **817**, and buttons **818**, have moved downward with the user's hand, and enlarged menu bar **603** (**903**) along with enlarged menu bar items **605** (**905**) have been pulled down from an offscreen position and are now viewable on screen **601**. As shown in FIG. **9**B, the user now has access to menu bar items **905**, and consequently, may be able to select items from drop-down menus of menu bar **903**. This may allow the user to access functions that might not otherwise be accessible in a low-resolution input mode UI.

[0046] Because the user can pull the menu bar back onto the screen in a similar manner as it was pushed off the screen, and because the menu bar appears at the same relative location, this gesture can be simple and intuitive for most users. When the user lets go of the screen, i.e., lifts his or her hand from the screen, the menu bar can automatically slide back off the screen as UI **800** returns to the view shown in FIG. **9**A.

[0047] The ability to un-obscure items can be desirable, particularly since many touch-based applications have fewer menu options and/or other functions than corresponding mouse-based applications, and it may be difficult and/or impractical to create a gesture for every possible function and feature in many of today's software applications. Therefore, compared to a mouse-based input UI for a particular application, a touch-based input UI may have scaled-back functionality, i.e., fewer buttons/selections/features/etc. However, the user may want to see and have access to those menus, scrollbars, etc., while still in a touch-based input mode, especially if the user wants to access a feature/button/menu item in the mouse-based input that is not included in the touch-based input.

[0048] In other embodiments, the process of transition effect **700** can be accomplished in other ways. For example, items could be slid off the screen in any direction, at any speed, etc. Portions of certain "obscured" items might remain displayed on the screen, for example, a part of dock **607** along with some dock items **609** could remain on display **601** while the rest of the dock slides off the screen. Items remaining on the screen might be enlarged by the same fixed percentage, by different percentages, and/or some not enlarged at all. The relative positions and/or appearance of some or all of the items remaining on the screen might be changed or modified in ways other than an increase in size.

[0049] FIG. **10** illustrates example computing system **1000** that can include one or more of the embodiments of the invention described above. Computing system **1000** can include one or more panel processors **1002** and peripherals **1004**, and panel subsystem **1006**. Peripherals **1004** can include, but are not limited to, random access memory (RAM) or other types of memory or storage, watchdog timers and the like. Panel subsystem **1006** can include, but is not limited to, one or more sense channels **1008**, channel scan logic **1010** and driver logic **1014**. Channel scan logic **1010** can access RAM **1012**, autonomously read data from the sense channels and provide control for the sense channels. In addition, channel scan logic **1010** can control driver logic **1014** to generate stimulation signals **1016** at various frequencies and phases that can be selectively applied to drive lines of touch sensor panel **1024**. In some embodiments, panel subsystem **1006**, panel processor **1002** and peripherals **1004** can be integrated into a single application specific integrated circuit (ASIC).

[0050] Touch sensor panel **1024** can include a capacitive sensing medium having a plurality of drive lines and a plurality of sense lines, although other sensing media can also be used. Each intersection of drive and sense lines can represent a capacitive sensing node and can be viewed as picture element (pixel) **1026**, which can be particularly useful when touch sensor panel **1024** is viewed as capturing an "image" of touch. (In other words, after panel subsystem **1006** has determined whether a touch event has been detected at each touch sensor in the touch sensor panel, the pattern of touch sensors in the multi-touch panel at which a touch event occurred can be viewed as an "image" of touch (e.g. a pattern of fingers touching the panel).) Each sense line of touch sensor panel **1024** can drive sense channel **1008** (also referred to herein as an event detection and demodulation circuit) in panel subsystem **1006**.

[0051] Computing system **1000** can also include host processor **1028** for receiving outputs from panel processor **1002** and performing actions based on the outputs that can include, but are not limited to, transitioning between a high-resolution input mode and a low-resolution input mode according to one or more embodiments of the invention, moving an object such as a cursor or pointer, scrolling or panning, adjusting control settings, opening a file or document, viewing a menu, making a selection, executing instructions, operating a peripheral