MULTI-APPLICATION ENVIRONMENT

BACKGROUND

[0001] Conventional operating systems permit users to view multiple computing applications through windows. Each of these windows generally includes a frame having controls for interacting with the computing application as well as controls for selecting which window frame is primary or moving, sizing, or otherwise managing the layout of the window. These window frames, however, occupy portions of a display that might otherwise be dedicated to an application's content. Furthermore, managing the primacy or layout of these windows through these controls can be time-consuming and annoying to users.

SUMMARY

[0002] This document describes techniques and apparatuses enabling a multi-application environment. The multi-application environment described herein can present multiple applications without dedicating significant amounts of a display to window frames for the applications and/or requiring management of window frames, such as their size, location, or primacy on the display. In some embodiments, these techniques and apparatuses enable multi-application environments having combinations of immersive interfaces, windows-based interfaces, and a desktop treated as an immersive interface. Also, in some embodiments the techniques and apparatuses enable management of applications in the multi-application environment, such as to size and move interfaces within the environment. Further still, some embodiments enable management of previously-interacted-with applications that are not currently presented. Further, some embodiments of the techniques enable management of applications and their interfaces, whether currently displayed or not, through edge gestures or a user interface management menu.

[0003] This summary is provided to introduce simplified concepts that are further described below in the Detailed Description. This summary is not intended to identify essential features of the claimed subject matter, nor is it intended for use in determining the scope of the claimed subject matter. Techniques and/or apparatuses enabling a multi-application environment are also referred to herein separately or in conjunction as the "techniques" as permitted by the context, though techniques may include or instead represent other aspects described herein.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Embodiments enabling a multi-application environment are described with reference to the following drawings. The same numbers are used throughout the drawings to reference like features and components:

[0005] Fig. 1 illustrates an example system in which techniques enabling a multi-application environment can be implemented.

[0006] Fig. 2 illustrates an example method for enabling edge gestures, the edge gestures being approximately perpendicular to an edge in which the gesture begins.

[0007] Fig. 3 illustrates an example tablet computing device having a touch-sensitive display presenting an immersive interface.

[0008] Fig. 4 illustrates the example immersive interface of Fig. 3 along with example edges.

[0009] Fig. 5 illustrates the example immersive interface of Figs. 3 and 4 along with angular variance lines from a perpendicular line and a line from a start point to a later point of a gesture.

[0010] Fig. 6 illustrates the edges of the immersive interface shown in Fig. 4 along with two regions in the right edge.

[0011] Fig. 7 illustrates an application-selection interface presented by a system-interface module in response to an edge gesture made over the immersive interface and webpage of Fig. 3.

[0012] Fig. 8 illustrates an example method for enabling edge gestures including determining an interface to present based on some factor of the gesture.

[0013] Fig. 9 illustrates an example method enabling expansion of, or ceasing presentation of, a user interface presented in response to an edge gesture or presentation of another user interface.

[0014] Fig 10 illustrates a laptop computer having a touch-sensitive display having a windows-based email interface and two immersive interfaces.

[0015] Fig. 11 illustrates the interfaces of Fig. 10 along with two gestures having a start point, later points, and one or more successive points.

[0016] Fig. 12 illustrates the windows-based email interface of Figs. 10 and 11 along with an email handling interface presented in response to an edge gesture.

[0017] Fig. 13 illustrates the interfaces of Fig. 12 along with an additional-email-options interface presented in response to a gesture determined to have a successive point a preset distance from the edge.

[0018] Fig. 14 illustrates a method for switching back to a previously-interacted-with application using a queue.

[0019] Fig. 15 illustrates an example interaction order in which a user interacts with various applications.

[0020] Fig. 16 illustrates the immersive interface of Fig. 3 along with a thumbnail image of a user interface of a prior application.

[0021] Fig. 17 illustrates a method for switching back to a previously-interacted-with application, which may or may not use a queue.

[0022] Fig. 18 illustrates the immersive interface of Figs. 3 and 16, two progressive presentations, and two gesture portions.

[0023] Fig. 19 illustrates a method enabling a multi-application environment, including altering sizes of multiple immersive interfaces responsive to a single selection.

[0024] Fig. 20 illustrates the desktop computing device of Fig. 1 having a touch-sensitive display shown displaying a multi-application environment with two immersive interfaces divided by an interface divider region.

[0025] Fig. 21 illustrates the multi-application environment of Fig. 20 with sizes of the two immersive interfaces altered and the interface divider region moved.

[0026] Fig. 22 illustrates a method for displaying an immersive interface of an application in a region responsive to as little as one selection and at a size fully occupying the region.

[0027] Fig. 23 illustrates a current immersive interface fully occupying a multi-application environment having three regions.

[0028] Fig. 24 illustrates the multi-application environment of Fig. 23 with a reduced-size immersive interface instead of the current immersive interface of Fig. 23 and a second immersive interface.

[0029] Fig. 25 illustrates a method for managing a multi-application environment through a user interface.

[0030] Fig. 26 illustrates an example multi-application environment having primary and non-primary regions.

[0031] Fig. 27 illustrates the multi-application environment of Fig. 26 including a user interface management menu.

[0032] Fig. 28 illustrates a method enabling display of a desktop as an immersive interface within a multi-application environment.

[0033] Fig. 29 illustrates an example multi-application environment having a desktop immersive interface displaying windows-based interfaces and a taskbar along with an immersive interface displaying content.

[0034] Fig. 30 illustrates a method enabling content presentation and/or management of a multi-application environment.

[0035] Fig. 31 illustrates an example device in which techniques enabling a multi-application environment can be implemented.

DETAILED DESCRIPTION

Overview

[0036] This document describes techniques and apparatuses enabling a multi-application environment. The multi-application environment described herein can present multiple applications without dedicating significant amounts of a display to window frames for the applications and/or requiring management of window frames, such as their size, location, or primacy on the display. In some embodiments, these techniques and apparatuses enable multi-application environments having combinations of immersive interfaces, windows-based interfaces, and a desktop treated as an immersive interface. Also, in some embodiments the techniques and apparatuses enable management of applications currently presented in or not currently presented in the multi-application environment, such as to size and move interfaces within the environment and select previously-interacted-with applications that are not currently presented. This and other forms of management are enabled, in some embodiments, through edge gestures made over the multi-application environment or a user interface management menu.

[0037] These are but a few examples of many ways in which the techniques enable a multi-application environment, others of which are described below.

Example System

[0038] Fig. 1 illustrates an example system 100 in which techniques enabling a multi-application environment can be embodied. System 100 includes a computing device 102, which is illustrated with six examples: a laptop computer 104, a tablet computing device 106, a smart phone 108, a set-top box 110, a desktop computer 112, and a gaming device 114, though other computing devices and systems, such as servers and netbooks, may also be used.

[0039] Computing device 102 includes computer processor(s) 116 and computer-readable storage media 118 (media 118). Media 118 includes an operating system 120, windows-based mode module 122, multi-application environment module 124, system-

interface module 126, gesture handler 128, application manager 130, which includes or has access to application queue 132, manager 134, and one or more applications 136, each having one or more application user interfaces 138.

[0040] Computing device 102 also includes or has access to one or more displays 140 and input mechanisms 142. Four example displays are illustrated in Fig. 1. Input mechanisms 142 may include gesture-sensitive sensors and devices, such as touch-based sensors and movement-tracking sensors (*e.g.*, camera-based), as well as mice (free-standing or integral with a keyboard), track pads, and microphones with accompanying voice recognition software, to name a few. Input mechanisms 142 may be separate or integral with displays 140; integral examples include gesture-sensitive displays with integrated touch-sensitive or motion-sensitive sensors.

[0041] Windows-based mode module 122 presents application user interfaces 138 through windows having frames. These frames may provide controls through which to interact with an application and/or controls enabling a user to move and resize the window.

[0042] Multi-application environment module 124 provides an environment by which a user may view and interact with one or more of applications 136 through application user interfaces 138. Multi-application environment 124 may present one or more application user interfaces 138 in conjunction with windows-based mode module 122. Multi-application environment 124 may also or instead present one or more application user interfaces 138 as immersive interfaces.

[0043] An immersive interface, in some embodiments, presents content of, and enables interaction with, applications with little or no window frame, without a need for a user to manage a window frame's layout or primacy relative to other windows (*e.g.*, which window is active or up front), without a need for a user to manually size or position application user interfaces 138, and/or without presenting visual controls (e.g., permanent controls on a window frame or in a window obscuring content).

[0044] The multi-application environment enabled by multi-application environment module 124 can be, but is not required to be, hosted and/or surfaced without use of a windows-based desktop environment. Thus, in some cases multi-application environment module 124 presents a multi-application environment as an immersive environment, and thus without a window (even one without a substantial frame) and/or precludes usage of desktop-like displays (*e.g.*, a taskbar). Further still, in some embodiments this multi-application environment is similar to an operating system in that it is not closeable or

capable of being un-installed. While not required, in some cases this multi-application environment enables use of all or nearly all of the pixels of a display by applications within the multi-application environment.

[0045] Examples of multi-application environments, some of which include only immersive interfaces and some of which include a windows-based interface or a desktop treated as an immersive interface, are provided below, though they are not exhaustive or intended to limit the techniques described herein.

[0046] System-interface module 126 provides one or more interfaces through which interaction with operating system 120 is enabled, such as an application-launching interface, a start menu, or a system tools or options menu, to name just a few.

[0047] Operating system 120, modules 122, 124, and 126, as well as gesture handler 128, application manager 130, and manager 134 can be separate from each other or combined or integrated in any suitable form.

Example Methods

[0048] Example methods 200, 800, and 900 address edge gestures, example methods 1400 and 1700 address switching back to a previously-interacted-with application, example methods 1900 and 2200 address managing an immersive interface in a multi-application environment, example method 2500 addresses managing a multi-application environment through a user interface, method 2800 address a desktop treated as an immersive interface within a multi-application environment, method 3000 addresses content presentation and/or management of a multi-application environment.

[0049] The methods may be used separately or in combination with each other, in whole or in part. For example, the techniques may use an edge gesture to enable selection, sizing, and switching of interfaces currently in a multi-application environment. The techniques may also select previously-interacted-with applications using an application queue, with or without an edge gesture. Further, the techniques may automatically size multiple interfaces on selection of a previously-interacted-with application through an application queue or responsive to a selection to move or alter a currently-displayed interface.

Edge Gestures

[0050] Fig. 2 depicts a method 200 for enabling edge gestures based on the edge gesture being approximately perpendicular to an edge in which the gesture begins. In portions of the following discussion, reference may be made to system 100 of Fig. 1 and other

methods and example embodiments described elsewhere herein, reference to which is made for example only.

[0051] Block 202 receives a gesture. This gesture may be received at various parts of a display, such as over a windows-based interface, an immersive interface, or no interface. Further, this gesture may be made and received in various manners, such as a pointer tracking a movement received through a touch pad, mouse, or roller ball or a physical movement made with arm(s), finger(s), or a stylus received through a motion-sensitive or touch-sensitive mechanism.

[0052] By way of example consider Fig. 3, which illustrates a tablet computing device 106. Tablet 106 includes a touch-sensitive display 302 shown displaying an immersive interface 304 that includes a webpage 306. As part of an ongoing example, at block 202 gesture handler 128 receives gesture 308 as shown in Fig. 3.

[0053] Block 204 determines whether a start point of the gesture is at an edge. As noted above, the edge in question can be an edge of a user interface, whether immersive or windows-based, and/or of a display. In some cases, of course, an edge of a user interface is also an edge of a display. The size of the edge can vary based on various factors about the display or interface. A small display or interface may have a smaller size in absolute or pixel terms than a large display or interface. A highly sensitive input mechanism permits a smaller edge as well. Example edges are rectangular and vary between one and twenty pixels in one dimension and an interface limit of the interface or display in the other dimension, though other sizes and shapes, including convex and concave edges may instead be used.

[0054] Continuing the ongoing example, consider Fig. 4, which illustrates immersive interface 304 and gesture 308 of Fig. 3 as well as left edge 402, top edge 404, right edge 406, and bottom edge 408. For visual clarity webpage 306 is not shown. In this example the dimensions of the interface and display are of a moderate size, between that of smart phones and that of many laptop and desktop displays. Edges 402, 404, 406, and 408 have a small dimension of twenty pixels, an area of each shown bounded by dashed lines at twenty pixels from the display or interface limit at edge limit 410, 412, 414, and 416, respectively. While shown overlapping at the corners, edges may instead be mitered at the corners or one edge can be favored over another (e.g., edge 404 is favored over edge 406 if a start point is received at the overlapping corner).

[0055] Gesture handler 128 determines that gesture 308 has a start point 418 and that this start point 418 is within left edge 402. Gesture handler 128 determines the start point

in this case by receiving data indicating [X,Y] coordinates in pixels at which gesture 308 begins and comparing the first of these coordinates to those pixels contained within each edge 402, 404, 406, and 408. Gesture handler 128 often can determine the start point and whether it is in an edge faster than a sample rate, thereby causing little or no performance downgrade from techniques that simply pass gestures directly to an exposed interface over which a gesture is made.

[0056] Returning to method 200 generally, if block 204 determines that the start point of the gesture is not at an edge, method 200 proceeds along a "No" path to block 206. Block 206 passes the gestures to an exposed user interface, such as an underlying interface over which the gesture was received. Altering the ongoing example, assume that gesture 308 was determined not to have a start point within an edge. In such a case gesture handler 128 passes buffered data for gesture 308 to immersive user interface 304. After passing the gesture, method 200 ends.

[0057] If block 204 determines that the start point of the gesture is in an edge, method 200 proceeds along a "Yes" path to block 208. Block 208 responds to the positive determination of block 204 by determining whether a line from the start point to a later point of the gesture is approximately perpendicular from the edge.

[0058] Block 208, in some embodiments, determines the later point used. Gesture handler 128, for example, can determine the later point of the gesture based on the later point being received a preset distance from the edge or the start point, such as past edge limit 410 for edge 402 or twenty pixels from start point 418, all of Fig. 4. In some other embodiments, gesture handler 128 determines the later point based on it being received a preset time after receipt of the start point, such an amount of time slightly greater than used generally by computing device 102 to determine that a gesture is a tap-and-hold or hover gesture.

[0059] For the ongoing embodiment, gesture handler 128 uses a later-received point of gesture 308 received outside of edge 402 so long as that later-received point is received within a preset time. If no point is received outside of the edge within that preset time, gesture handler 128 proceeds to block 206 and passes gesture 308 to immersive interface 304.

[0060] Using the start point, block 208 determines whether a line from the start point to the later point of the gesture is approximately perpendicular to the edge. Various angles of variance can be used in this determination by block 208, such as five, ten, twenty, or thirty degrees.

[0061] By way of example, consider an angle of variance of thirty degrees from perpendicular. Fig. 5 illustrates this example variance, showing immersive interface 304, gesture 308, left edge 402, left edge limit 410, and start point 418 of Figs. 3 and 4 along with a thirty-degree variance lines 502 from perpendicular line 504. Thus, gesture handler 128 determines that line 506 from start point 418 to later point 508 (which is at about twenty degrees from perpendicular) is approximately perpendicular based on being within the example thirty-degree variance line 502.

[0062] Generally, if block 208 determines that the line is not approximately perpendicular to the edge, method 200 proceeds along a "No" path to block 206. As noted in part above, block 208 may also determine that a later point or other aspect of a gesture disqualifies the gesture. Examples include when a later point is within the edge, such as due to a hover, tap, press-and-hold, or up-and-down gesture (*e.g.*, to scroll content in the user interface), or when the gesture is set to be a single-input gesture and a second input is received (*e.g.*, a first finger starts at an edge but a second finger then lands anywhere).

[0063] If block 208 determines that the line is approximately perpendicular based a later point outside the edge, method 200 proceeds along a "Yes" path to block 210.

[0064] Block 210 responds to the positive determination of block 208 by passing the gesture to an entity other than the exposed user interface. This entity is not a user interface over which the gesture was received, assuming it was received over a user interface at all. Block 210 may determine to which entity to pass the gesture as well, such as based on an edge or region of an edge in which the start point of the gesture is received. Consider Fig. 6, for example, which illustrates immersive interface 304 and edges 402, 404, 406, and 408 of Fig. 4 but adds top region 602 and bottom region 604 to right edge 406. A start point in top region 602 can result in a different entity (or even a same entity but a different user interface provided in response) than a start point received to bottom region 604. Likewise, a start point in top edge 404 can result in a different entity or interface than left edge 402 or bottom edge 408.

[0065] In some cases, this entity is an application associated with the user interface rather than an application not associated with the user interface, such as a system entity or a different application. In such a case, passing the gesture to the entity can be effective to cause the application to present a second user interface enabling interaction with the application. In the movie example above, the entity can be the media player playing the movie but not the immersive interface displaying the movie. The media player can then present a second user interface enabling selection of subtitles or a director's commentary

rather than selections enabled by the interface displaying the movie, such as "pause," "play," and "stop." This capability is permitted in Fig. 1, where one of applications 136 can include or be capable of presenting more than one application user interface 138. Thus, block 210 can pass the gesture to system-interface module 126, the one of applications 136 currently presenting the user interface, or another of applications 136, to name just three possibilities.

[0066] Concluding the ongoing embodiment, at block 210 gesture handler 128 passes gesture 308 to system-interface module 126. System-interface module 126 receives the buffered portion of gesture 308 and continues to receive the rest of gesture 308 as it is made by the user. Fig. 7 illustrates a possible response upon receiving gesture 308, showing an application-selection interface 702 presented by system-interface module 126 and over immersive interface 304 and webpage 306 from Fig. 3. Application-selection interface 702 enables selection of various other applications and their respective interfaces at selectable application tiles 704, 706, 708, and 710.

[0067] The example application-selection interface 702 is an immersive user interface presented using multi-application environment module 124, though this is not required. Presented interfaces may instead be windows-based and presented using windows-based mode module 122. Both of these modules are illustrated in Fig. 1.

[0068] Block 210 may also or instead determine to pass the gesture to different entities and/or interfaces based on other factors about the gesture received. Example factors are described in greater detail in method 800 below.

[0069] Note that method 200 and other methods described hereafter can be performed in real-time, such as while a gesture is being made and received. This permits, among other things, a user interface presented in response to a gesture to be presented prior to completion of the gesture. Further, the user interface can be presented progressively as the gesture is received. This permits a user experience of dragging out the user interface from the edge as the gesture is performed with the user interface appearing to "stick" to the gesture (*e.g.*, to a mouse pointer or person's finger making the gesture).

[0070] Fig. 8 depicts a method 800 for enabling edge gestures including determining an interface to present based on some factor of the gesture. In portions of the following discussion reference may be made to system 100 of Fig. 1, reference to which is made for example only. Method 800 may act wholly or partly separate from or in conjunction with other methods described herein.

[0071] Block 802 determines that a gesture made over a user interface has a start point at an edge of the user interface and a later point not within the edge. Block 802 may operate similarly to or use aspects of method 200, such as determining a later point on which to base block 802's determination. Block 802 may act differently as well.

[0072] In one case, for example, block 802 determines that a gesture is a single-finger swipe gesture starting at an edge of an exposed immersive user interface and having a later point not at the edge but not based on an angle of the gesture. Based on this determination, block 802 proceeds to block 804 rather than pass the gesture to the exposed immersive user interface.

[0073] Block 804 determines which interface to present based on one or more factors of the gesture. Block 804 may do so based on a final or intermediate length of the gesture, whether the gesture is single or multi-point (*e.g.*, a single-finger or multi-finger gesture), or a speed of the gesture. Thus, block 804 may determine to present a start menu in response to a multi-finger gesture, an application-selection interface in response to a relatively short single-finger gesture, or a system-control interface permitting selection to shut down computing device 102 in response to relatively long single-finger gesture, for example. To do so, gesture handler 128 may determine the length of the gesture or a number of inputs (*e.g.*, fingers). In response, block 806 presents the determined user interface.

[0074] Assume, by way of example, that gesture handler 128 determines, based on a factor of the gesture, to present a user interface enabling interaction with operating system 120. In response system-interface module 126 presents this user interface. Presentation of the user interface can be similar to manners described in other methods, such as with a progressive display of application-selection user interface 702 of Fig. 7.

[0075] Following method 200 and/or method 800 in whole or in part, the techniques may proceed to perform method 900 of Fig. 9. Method 900 enables expansion of a user interface, presentation of another interface, or ceasing presentation of the user interface presented in response to an edge gesture.

[0076] Block 902 receives a successive point of the gesture and after presentation of at least some portion of the second user interface. As noted in part above, methods 200 and/or 800 are able to present or cause to be presented a second user interface, such as a second user interface for the same application associated with a current user interface, a different application, or a system user interface.

[0077] By way of example, consider Fig. 10, which illustrates a laptop computer 104 having a touch-sensitive display 1002 displaying a windows-based email interface 1004 and two immersive interfaces 1006 and 1008. Windows-based email interface 1004 is associated with an application that manages email, which can be remote or local to laptop computer 104. Fig. 10 also illustrates two gestures, 1010 and 1012. Gesture 1010 proceeds in a straight line while gesture 1012 reverses back (shown with two arrows to show two directions).

[0078] Fig. 11 illustrates gesture 1010 having a start point 1102, a later point 1104, and a successive point 1106, and gesture 1012 having a same start point 1102, a later point 1108, and a first successive point 1110, and a second successive point 1112. Fig. 11 also shows a bottom edge 1114, a later-point area 1116, and an interface-addition area 1118.

[0079] Block 904 determines, based on the successive point, whether the gesture includes a reversal, an extension, or neither. Block 904 may determine a reversal by determining that a successive point is at the edge or is closer to the edge than a prior point of the gesture. Block 904 may determine that the gesture extends based on the successive point being a preset distance from the edge or the later point. If neither of these is determined to be true, method 900 may repeat blocks 902 and 904 to receive and analyze additional successive points until the gesture ends. If block 904 determines that there is a reversal, method 900 proceeds along "Reversal" path to block 906. If block 904 determines that the gesture is extended, method 900 proceeds along an "Extension" path to block 908.

[0080] In the context of the present example, assume that gesture handler 128 receives first successive point 1110 of gesture 1012. Gesture handler 128 then determines that first successive point 1110 is not at edge 1114, is not closer than a prior point of the gesture to edge 1114 (*e.g.*, is not closer than later point 1108), and is not a preset distance from the edge or later point by not being within interface-addition region 1118. In such a case method 900 returns to block 902.

[0081] On a second iteration of block 902, assume that gesture handler 128 receives second successive point 1112. In such a case, gesture handler 128 determines that second successive point 1112 is closer to edge 1114 than first successive point 1110 and thus gesture 1012 includes a reversal. Gesture handler 128 then proceeds to block 906 to cease to present the second user interface previously presented in response to the gesture. By way of example, consider Fig. 12, which illustrates an email handling interface 1202. In

this example case of block 906, gesture handler 128 causes the email application to cease to present interface 1202 in response to a reversal of gesture 1012 (not shown removed).

[0082] Block 908, however, presents or causes presentation of a third user interface or expansion of the second user interface. Continuing the ongoing example, consider Fig. 13, which illustrates additional-email-options interface 1302 in response to gesture 1010 determined to have successive point 1106 a preset distance from edge 1104, in this case being within interface-addition region 1118 of Fig. 11. This region and preset distance can be set based on a size of the user interface previously presented in response to the gesture. Thus, a user wishing to add additional controls may simply extend the gesture past the user interface presented in response to an earlier portion of the gesture.

[0083] Method 900 can be repeated to add additional user interfaces or expand a presented user interface. Returning to the example interface 702 of Fig. 7, for example, gesture handler 128 can continue to add interfaces or controls to interface 702 as gesture 308 extends past interface 702, such as by presenting an additional set of selectable application tiles. If gesture 308 extends past the additional tiles, gesture handler 128 may cause system-interface module 126 to present another interface adjacent the tiles to enable the user to select controls, such as to suspend, hibernate, switch modes (immersive to windows-based and the reverse), or shut down computing device 102.

[0084] While the above example user interfaces presented in response to an edge gesture are opaque, they may also be partially transparent. This can be useful by not obscuring content. In the movie example described above, a user interface presented can be partially transparent thereby permitting the movie to be only partially obscured during use of the user interface. Similarly, in the example of Figs. 12 and 13, interfaces 1202 and 1302 may be partially transparent, thereby enabling a user to see the text of the email while also selecting a control in one of the interfaces.

[0085] As noted above, example methods 200, 800, and 900 address edge gestures and are described prior to methods 1400 and 1700, which address switching back to a previously-interacted-with application. Any one or more of the method may be used separately or in combination with, in whole or in part, others of the methods.

Switching Back to a Previously-Interacted-With Application

[0086] Fig. 14 depicts a method 1400 for switching back to a previously-interacted-with application using a queue. In portions of the following discussion reference may be made to system 100 of Fig. 1 and other methods and example embodiments described elsewhere herein, reference to which is made for example only.

[0087] Block 1402 maintains a queue of multiple interacted-with applications, the queue arranged by most-recently-interacted-with to least-recently-interacted-with applications other than a current application. Consider, for example, Fig. 15, which illustrates an interaction order 1502 in which a user interacts with various applications. First, the user interacts with a web-searching application 1504 through its interface. Second, the user interacts with a web-enabled media application 1506 through a web browser. Third, the user interacts with a local (non-web) photo application 1508 through its interface. Fourth, the user interacts with a social-networking application 1510 through the web browser. Fifth, the user returns to interacting with the web-enabled media application 1506. Sixth, the user interacts with a web-enabled news application 1512 again through the web browser.

[0088] For the first interaction no queue is maintained as no other applications have been interacted with prior to this first interaction. For the second through sixth interactions of interaction order 1502, consider queues 1514, 1516, 1518, 1520, and 1522, which correspond to each interaction in interaction order 1502 after the first interaction, respectively. Queues 1514 to 1522 are example iterations of application queue 132 maintained by application manager 130, both of Fig. 1.

[0089] As shown in Fig. 15, application manager 130 keeps application queue 132 up-do-date based on a user's interactions. Queue 1522, for example, includes media application 1506 as the most-recently-interacted application, followed by social-networking application 1510, photo application 1508, and ending with web-searching application 1504. As the user interacts with media application 1506 twice (at the second and fifth interaction) application manager 130 removes it from application queue 132 at the fifth interaction and reorders the other applications to reflect an up-to-date order of interactions but excluding currently-interacted-with applications.

[0090] Block 1404 receives a gesture or gesture portions. This gesture or gesture portions can include one or more of the various gestures or portions described elsewhere herein, such as a pointer tracking a movement received through a touch pad, mouse, or roller ball or a physical movement made with arm(s), finger(s), or a stylus received through a motion-sensitive or touch-sensitive mechanism. In some embodiments, gesture portions are received, each portion being part of one gesture and each resulting in presentation of an application in the queue. Each of these portions may have, but are not required to have, a start point at an edge of a display, a later point not at the edge of the display, and a successive point at the edge of the display. A gesture having multiple

portions in this case would look something like a multi-loop spiral, multiple circles, or a back-and-forth (*e.g.*, zigzag) where each loop, circle, or back-and-forth starts, leaves, and returns to an edge of a user interface or display. Optionally, block 1404 may receive a number of gestures or gesture portions. These gestures or gesture portions can include one or more of the various gestures or gestures portions described elsewhere herein.

[0091] Continuing the ongoing embodiment, consider again Fig. 3, which illustrates tablet computing device 106 having touch-sensitive display 302 shown displaying immersive interface 304 including webpage 306. For this example, assume that immersive interface 304 is associated with news application 1512 and that webpage 306 is content from news application 1512.

[0092] As part of this example, at block 1404, gesture handler 128 receives gesture 308 as shown in Fig. 3, which gesture handler 128 passes to application manager 130. For the ongoing example, assume that gesture 308 is determined to be associated with switching back to a previously-interacted-with application rather than some other function or application.

[0093] Block 1406, responsive to receiving the gesture or gesture portions, proceeds through the queue to another application of the multiple interacted-with applications. Thus, on receiving the gesture or gesture portion(s), application manager 130 may proceed to the first, and thus the most-recently-interacted-with of the applications of application queue 132. In some embodiments, on receiving two gestures or portions, application manager 130 may proceed to the second most-recently-interacted-with application of application queue 132, though method 1400 may do so by repeating blocks 1404, 1406 and/or 1408, and so forth as described below.

[0094] Continuing the ongoing embodiment, assume that gesture 308 is received after the sixth interaction at which time the currently-interacted-with application is news application 1512 and that application queue 132 is up-to-date and represented by queue 1522 of Fig. 15. In such a case, application manager 130 proceeds to media application 1506 on receiving the gesture or gesture portion.

[0095] Block 1408 presents a user interface associated with the other application. This user interface, in some embodiments, is the same user interface through which interaction with the application was previously made. In some embodiments, the user interface is presented as a thumbnail or transparent overlay above the currently presented user interface. Application manager 130 presents this user interface alone or in combination

with the associated application, such as by causing the associated application to present the user interface with which the user last interacted.

[0096] For this example, application manager 130 presents a thumbnail image of the user interface for the application progressively as gesture 308 is received and then expands the thumbnail to encompass the available real estate of the display when the gesture ends. Application manager 130 thereby replaces webpage 306 in immersive interface 304 or replaces immersive interface 304 with another interface, which can be immersive or windows-based.

[0097] This is illustrated in Fig. 16 with thumbnail image 1602 of a user interface of media application 1506 presented over immersive interface 304 and webpage 306 of news application 1512. After gesture 308 ends, thumbnail image 1602 expands into media player 1604, replacing webpage 306 in immersive interface 304. This is but one example manner for presenting the user interface for the selected application, others manners for responding progressively or otherwise are described elsewhere herein.

[0098] In some embodiments, block 1408 shrinks the current user interface to a second thumbnail image and passes the second thumbnail image toward a region of a display from which the first-mentioned thumbnail image is progressively presented. Thus, block 1408 expands thumbnail image 1602 into media player 1604 while shrinking webpage 306 to a thumbnail image and passing that thumbnail to the edge from which thumbnail image 1602 was selected.

[0099] During the presentation of the user interface at block 1408, another gesture or gesture portion may be received, returning to block 1404. In some cases, the other gesture or gesture portion is received within an amount of time while the user interface is presented by block 1408. Following the return to block 1404, block 1406 may then proceed to yet another or subsequent application of the multiple interacted-with applications. Continuing this progression, block 1408 then presents a user interface associated with the subsequent application of the multiple interacted-with applications.

[00100] Thus, by repeating blocks 1404, 1406, and 1408 user interfaces associated with previously interacted-with applications can be successively presented. In some cases, a user interface associated with a previously-interacted with application can be presented responsive to each gesture received. In the context of the present example, when another gesture is received while presenting the user interface of media application 1506, a user interface associated with social-networking application 1510 (the second most-recently interacted with application of queue 1522) is presented. Receiving yet another gesture or

gesture portion during the presentation of the user interface associated with social-networking application 1510 results in a presentation of a user interface associated with photo application 1508 (the third most-recently interacted with application of queue 1522) and so forth.

[00101] Following this switch from presenting a current application to a presenting another selected, prior application, block 1410 updates the queue responsive to interaction with, or a time period passing during presentation of, the user interface associated with the other application. In some cases a prior application may be selected and then another quickly selected after it, effectively a scanning through of the applications in the queue. In such cases, block 1410 may forgo updating the queue, as a quick viewing may not be considered an interaction.

[00102] Example interactions with which application manager 130 updates application queue 132 include an explicit selection to interact with the newly presented interface, such as to control playback or edit information relating to currently playing media using controls shown in the user interface of media player 1604 of Fig. 16. In other cases an interaction is determined based on a time period passing. Assume, for example, that the news application's webpage is presented on selection rather than being the current application. After some period, such as one, two, or three seconds, for example, application manager 130 determines that the delay is effectively an interaction based on a likelihood that the user is reading the news article in the webpage. Similarly, presentation of a user interface for a media application at block 1408 that is playing media and remains on the display without another selection of applications in application queue 132 can also be considered an interaction.

[00103] As noted in part above, application queue 132 can be circular. In so doing, selection of applications is not stopped but rather rolls if a user reaches a least-recently-interacted with application of application queue 132. For example, on selecting to switch back to a prior application from social-networking application 1510 and thus using queue 1518, switching back once results in selecting photo application 1508, twice results in media application 1506, and three times to web-searching application 1504. A fourth selection to switch back returns, in a circular fashion, to again result in presenting photo application 1508.

[00104] Method 1400 describes various ways in which the techniques can enable selection of previously-interacted-with applications and determine which to present based on a queue. Method 1700 may operate in conjunction with method 1400 and other

methods described herein, though using a queue is not required. Therefore, method 1400 is not intended to limit the techniques as described in example method 1700.

[00105] Fig. 17 depicts a method 1700 for switching back to a previously-interacted-with application, which may or may not use a queue. In portions of the following discussion reference may be made to system 100 of Fig. 1, methods 200, 800, 900, 1400, and example embodiments described above, reference to which is made for example only.

[00106] Block 1702 enables selection of a previously-interacted-with application through a gesture made over a current user interface associated with a current application. Block 1702 may do so in various manners described above, such as with an edge gesture or portion thereof, as but one example.

[00107] Block 1704, responsive to receiving the gesture and without further selection, presents a previous user interface associated with the previously-interacted-with application.

[00108] Assume, for example, that a portion of a gesture is received associated with selection of a prior application, such as an edge gesture starting at an edge of the current user interface and proceeding approximately perpendicularly away from the edge. In response, block 1704 presents the user interface for the previously-interacted-with application or a thumbnail image of the interface, or some indicator that selection has successfully been made along with an indicator of the application or the interface selected.

[00109] Example thumbnail images or indicators include any of selectable application tiles 704, 706, 708, and 710 of Fig. 7 some of which include a thumbnail image of an interface while others indicate the application selected. Another example is thumbnail image 1602 of Fig. 16.

[00110] Block 1704 presents the user interface of the selected, previously-interacted-with application, as shown in Fig. 16 at media player 1604. In so doing, block 1704 may enable interaction with photo application 1508 through immersive interface 304 without further selection. Thus, a user after selecting, with as little as one gesture, a prior application may interact without needing to make another selection. The user need not select to exit an application-selection mode, for example, or make the presented interface "live" or primary or on top of the stack. Simply put, the techniques enable selection of a prior application and further interaction with that prior application with a single input.

[00111] In this example of Fig. 16, immediately after media player 1604 is presented and replaces webpage 306, a next input to immersive interface 304 is passed immediately to photo application 1508. Thus, a tap, hot key, or other input is passed directly to photo

application 1508, thereby enabling an immediate response by photo application 1508 to the input.

[00112] In some embodiments, the gesture made over the current user interface includes portions, each of which indicates a selection of a prior application. In such a case, block 1704 presents the previous user interface in response to the first portion and then, responsive to block 1702 receiving the second portion of the gesture, presents a further-previous user interface associated with a further previously-interacted-with application, and so forth.

[00113] This is illustrated in Fig. 18, which presents immersive interface 304 of Fig. 16 (shown twice for visual clarity), and ways in which block 1704 can respond to multiple gestures or portions of a single gesture. Fig. 18 illustrates two progressive presentations, 1802 and 1804, and gesture 1806 having two gesture portions 1806-1 and 1806-2, respectively. First progressive presentation 1802 illustrates a drag from a left edge of immersive interface 304 of thumbnail image 1602, and thus selection of the previously-interacted with photo application 1508. Note that thumbnail image 1602 "sticks" to gesture portion 1806-1. Note also that gesture 1806, unlike gesture 308 of Figs. 3 and 16, returns to the left edge. In response, rather than gesture 308 ending and media player 1604 replacing webpage 306, gesture portion 1806-1 of gesture 1806 returns to the edge at which it began. In this case thumbnail image 1602 is progressively displayed with gesture portion 1806-1 but then disappears when gesture portion 1806-1 returns to the edge.

[00114] Gesture 1806 continues with second portion 1806-2. In response, block 1704 presents second progressive presentation 1804, illustrating a second drag from the left edge of immersive interface 304. Here a social network thumbnail image 1808 of a further prior application, social-networking application 1510, is progressively presented. Gesture 1806 returns to the left edge as part of second portion 1806-2. In response, block 1704 drops off thumbnail image 1808 when gesture portion 1806-2 returns to the edge. This is but one example of ways in which the techniques enable users to select and view prior applications, even all of the previously-interacted-with applications, with only a single gesture. At any point in this example, gesture 1806 may end or indicate selection to present the full user interface for the selected application, at which time block 1704 presents the user interface (*e.g.*, media player 1604 of Fig. 16 or a full user interface for the social-networking application).

[00115] As noted above, example methods 200, 800, and 900 address edge gestures and are described prior to methods 1400 and 1700, which address switching back to a

previously-interacted-with application, which are in turn described prior to methods 1900 and 2200. Any one or more of the method may be used separately or in combination with, in whole or in part, others of the methods.

Managing an Immersive Interface

[00116] Fig. 19 depicts a method 1900 enabling a multi-application environment, including altering sizes of multiple immersive interfaces responsive to a single selection. In portions of the following discussion reference may be made to system 100 of Fig. 1 and other methods and example embodiments described elsewhere herein, reference to which is made for example only.

[00117] Block 1902 enables selection to alter a first size of a first immersive interface of a first application displayed in a multi-application environment in which a second immersive interface of a second application is displayed at a second size.

[00118] Block 1902 can enable this selection in various manners set forth above, such as with a gesture, whether made through a gesture-sensitive display or a track pad or mouse, or with a hardware button or hot keys, to name just a few.

[00119] Consider, by way of example, a case where block 1902 enables a select-and-move gesture selection through a gesture-sensitive display, the select-and-move gesture of an interface divider region between immersive interfaces of a multi-application environment. This example is illustrated in Fig. 20, which illustrates a desktop computing device 112 having a touch-sensitive display 2002 shown displaying a multi-application environment 2004. Multi-application environment 2004 includes a larger immersive interface 2006 and a smaller immersive interface 2008 separated by an immersive interface divider 2010. Larger immersive interface 2006 is associated with a word-processing application and presents document content 2012. Smaller immersive interface 2008 is associated with a software mapping application and presents mapping content 2014. As part of an ongoing example, at block 1902 manager 134 receives gesture 2016 as shown in Fig. 20, shown with an arrow but omitting an input actor (*e.g.*, a finger or stylus).

[00120] Block 1904, responsive to selection to alter the first size of the first immersive interface, alters the first size of the first immersive interface and the second size of the second immersive interface. Block 1904, therefore, may alter sizes of multiple immersive interfaces responsive to as few as one selection. Further, block 1904 may do so concurrently and without occluding either of the interfaces.

[00121] By way of example, consider the ongoing example of Fig. 20. Responsive to select-and-move gesture 2016 of interface divider region 2010, manager 134 reduces one interface and increases the other concurrently, here increasing smaller immersive interface 2008 and decreasing, at the same time, larger immersive interface 2006. The result of this alteration is illustrated in Fig. 21 at altered smaller immersive interface 2102 and altered larger immersive interface 2104. The prior position of interface divider region 2010 is shown at prior position 2106. Note also that select-and-move gesture 2016 starts at prior position 2106 of interface divider region 2010 and ends at final position 2108 of interface divider region 2010.

[00122] Note that in this example, multi-application environment 2004 is fully occupied with the immersive interfaces, both prior to and after altering sizes of the immersive interfaces, without unused real estate or real estate occluded with visible controls for managing the immersive interfaces.

[00123] This particular example illustrates one way in which the techniques permit a user to select sizes of immersive interfaces, here to increase a map presented by the mapping application.

[00124] The techniques also permit users to "snap" immersive interfaces to automatically fill a predetermined region of multi-application environment 2004. By so doing, gestures and other selections can be used that are fast and easy for users. Further, these regions can have a predetermined size across multiple devices, thereby permitting application developers to prepare for the region sizes. This is especially useful for smaller region sizes, as smaller sizes are often more challenging to present in a user-friendly manner. Consider again Fig. 20, for example, which illustrates a predetermined small-region width 2018, here having a width of 320 pixels. In this example, three regions are shown, two of which are subordinate in that they are included within a full region. These regions have widths, width 2018 and remainder width 2020 for the two subordinate regions, and a full width 2022 of a full region, all of multi-application environment 2004. Note that remainder width 2020 can vary across displays, as can full width 2022.

[00125] Block 1902 may also enable selection through a drag-and-drop gesture of one of the immersive interfaces from one region to another region. In such a case block 1904 may switch the interfaces between the regions or automatically move a divider (*e.g.*, immersive interface divider 2010 of Fig. 20) such that resulting sizes are switched. By so doing, manager 134 automatically reduces larger immersive interface 2006 to fully occupy a region previously occupied by smaller immersive interface 2008 and vice-versa.

[00126] In some cases selection to alter a size of an interface is enabled through an edge gesture. Consider, for example, an edge gesture starting at an edge of larger immersive interface 2006 and having a later point not at the edge of larger immersive interface 2006. Manager 134, alone or in conjunction with gesture handler 128 and/or application manager 130, shrinks larger immersive interface 2006 to a reduced-size image. Selection to resize interface 2006, then, can be performed by dropping the reduced-size image over smaller immersive interface 2008. In response, manager 134 resizes both interfaces.

[00127] Method 1900 describes various ways enabling a multi-application environment, including altering sizes of multiple immersive interfaces responsive to a single selection. Method 2200 may operate in conjunction with method 1900 and other methods described herein, though using a queue is not required. Therefore, method 1900 is not intended to limit the techniques as described in example method 2200.

[00128] Fig. 22 depicts a method 2200 for displaying an immersive interface of an application in a region, including responsive to as little as one selection and at a size fully occupying the region. In portions of the following discussion reference may be made to system 100 of Fig. 1, methods 200, 800, 900, 1400, 1700, and 1900, and example embodiments described above, reference to which is made for example only.

[00129] Block 2202 enables selection to display an immersive interface of an application in one of multiple regions of a multi-application environment displaying one or more current immersive interfaces of one or more current applications. Block 2202 may do so in various manners described above, such as with an edge gesture or portion thereof, as but one example. Further, the application selected can be a previously-interacted with application determined in various manners, such as by application manager 130 using application queue 132, both of Fig. 1.

[00130] The multi-application immersive interface can, at block 2202, present one, two, or even three current immersive interfaces. Thus, block 2202 permits selection of an application to place in regions currently occupied or that exist but are occupied by a larger immersive interface, such as in cases where one immersive interface fully occupies a multi-application environment.

[00131] By way of example, consider Fig. 23, which illustrates a current immersive interface 2302 occupying multi-application environment 2304. Note here that there are three subordinate regions, 2306, 2308, and 2310. These regions may be indicated or not. In cases where an application has been selected and is hovered or moved over one of the regions, the region can be indicated with partially transparent immersive interface dividers

2312 and 2314. These three subordinate regions 2360, 2308, and 2310 are included within full-size region 2316 occupying substantially all of multi-application environment 2304.

[00132] By way of example, assume that manager 134 receives a previously-interacted-with application selected according to method 1700 and following the example illustrated in Fig. 18. In such as case, assume that thumbnail image 1808 for social-networking application 1510 is selected and hovered over region 2306 (not shown but similar to Fig. 18). In response, manager 134 indicates that region 2306 is or is about to be selected and the size of region 2306 by displaying partially transparent immersive interface divider 2312.

[00133] Returning to method 2200, block 2204, responsive to the selection to display the immersive interface in the region, displays the immersive interface at a size fully occupying the region. Note that the user, with a little as the one selection of the application, can select and have presented an immersive interface at a size fully occupying a selected region.

[00134] Continuing the example, consider Fig. 24, which illustrates multi-application environment 2304 but now with a reduced-size immersive interface 2402 instead of current immersive interface 2302 of Fig. 23, and with a second immersive interface 2404 showing a social-networking webpage 2406 for social-networking application 1510 of Fig. 15. Second immersive interface 2404 fully occupies region 2306 and without user selection other than selection of the region.

[00135] Note that the arrangement of content in reduced-size immersive interface 2402 and social-networking webpage 2406 are both changed. Size changes can be made more quickly or allow for better content arrangements applications and/or developers of those applications having these region sizes in advance, which are provided by the techniques as predetermined region widths. Here the predetermined region width provided is region 2306, though a fill-width region 2408 may also be provided.

[00136] Following block 2204, method 2200 may repeat blocks 2202 and 2204, thereby enabling selection of additional immersive interfaces. For example, manager 134 can enable selection of a third immersive interface for presentation in region 2310 or 2308 of Fig. 23. In response to such a selection, manager 134 reduces the size of, or replaces, reduced-size immersive interface 2402.

[00137] Note that any of these methods may be combined in whole or in part. Thus, a gesture portion, for example, may be used to select an immersive interface and another portion of the same gesture select to place and/or size the immersive interface. In

response to this single gesture, the techniques can resize multiple interfaces currently presented in a multi-application environment.

User Interface for Managing a Multi-Application Environment

[00138] Fig. 25 depicts a method 2500 for managing a multi-application environment through a user interface. In portions of the following discussion reference may be made to system 100 of Fig. 1 and other methods and example embodiments described elsewhere herein, reference to which is made for example only.

[00139] Block 2502 enables selection of a user interface for managing a multi-application environment. Selection of the user interface can be made in various ways including those described above, such as with a gesture or portion of a gesture, hardware button or hot key, or voice command, just to name a few. Prior to selection, the user interface can be fully displayed, partially displayed, or not at all displayed. Consider, for example, a scenario where a multi-application environment is displayed and the user interface is not displayed. An example of such a scenario is illustrated in Fig. 26, which shows display 2600 filled with multi-application environment 2602. Multi-application environment 2602 includes a primary region 2604 and a non-primary region 2606, both of which present various content from applications 136. Note that non-primary region 2606 includes two non-primary sections 2608 and 2610, each of which may be used to present content in parallel with each other and that of primary region 2604. In this example, content from three applications is presented in parallel: content 2612 from a social networking website, content 2614 from a news website, and content 2616 from a local document-viewing application.

[00140] Here manager 134 of Fig. 1 enables selection of the user interface with a non-visual selector, such as a hot key or gesture (e.g., an edge gesture made to a right edge of multi-application environment 302). In some other cases, however, manager 134 enables selection through a displayed, selectable control.

[00141] Block 2504 displays the user interface responsive to this selection. The user interface can be an immersive user interface or a semi-transparent overlay enabling further selection. Through this user interface, block 2506 enables selection of an application for parallel presentation in the multi-application environment.

[00142] This user interface enables a user to manage the multi-application environment, though it may enable this in various manners. The user interface, for example, may enable a user to present, remove, or set as default an application to the user interface and/or the environment, such as through selection of labels associated with, or representations of, the

applications. The user interface may enable a user to select preferences for applications to be presented in the user interface, alter the environment generally, or switch to a non-multi-application environment. Furthermore, the user interface may present applications for selection based on various criteria, such as those applications that are most-recently used or most-often used by a user of computing device 102, as well as whether or not the application is currently executing. Further still, the user interface may present a common set of system commands pertaining to an application, such as a user interface that enables search commands, sharing content, or altering settings.

[00143] Continuing the ongoing example, assume that manager 134 receives, through a gesture made on a touch screen of display 2600, a selection to present the user interface. In response, manager 134 presents user interface management menu 2700 illustrated in Fig. 27, through which selection of applications is enabled. This example user interface management menu 2700 presents icons and/or names for nine applications. The nine applications include various websites, services, and local computing applications, and are named "Social Net," "News.com," "PDFs," "Telecon," "Music.com," "Movies.com," "Poker.com," "Art Space,", and "Maps by GPS" at 2702, 2704, 2706, 2708, 2710, 2712, 2714, 2716, and 2718, respectively. As noted, other applications, such as those pertaining to system commands, can also be presented by manager 134. By way of example, consider system commands 2719 shown as a tenth application of user interface management menu 2700, named "Search Share Settings."

[00144] In this particular example, two applications, 2702 and 2704, are "pinned." Pin icons, shown at 2720 and 2722, indicate that applications 2702 and 2704 will be maintained in one or both of multi-application environment 2602 and in user interface management menu 2700 (here it indicates both). Thus, both the "Social Net" and "News.com" applications will be executing and presenting content within some part of environment 2602 without further selection by the user (shown in sections 2608 and 2610, respectively). Note that selection of maintained ("pinned") applications can be enabled by manager 134 in various ways, such as through user interface management menu 2700 presenting pinning selection icon 2724 for selection of any of applications 2702-2718. As noted, any of the selectable applications may or may not be currently executing – in this case applications 2702, 2704, 2706, and 2708 are executing and applications 2710-2718 are not currently executing.

[00145] User interface management menu 2700 may also present applications based on other criteria, such as a user's history. Applications 2706, 2708, 2710, and 2712, for

example, are presented based on being the four most-recently-used by a user of computing device 102 (other than 2702 and 2704). Applications 2714, 2716, and 2718 are presented based on being most-often-used, excluding applications 2702-2712. In other cases, applications may be presented based on new content associated with the application being available (*e.g.*, new emails, messages, or RSS feeds) or as other alerts for the application are received.

[00146] Further still, user interface management menu 2700 enables a user to manage multi-application environment 2602 in addition to management of applications, such as with window selection icon 2726. Through this selection, manager 134 permits a user to select to exit multi-application environment 2602 and continue with currently-presented content using a windows-based environment.

[00147] Returning to method 2500, block 2508, responsive to selection of one or more applications, causes content of a selected application to be presented in the multi-application environment in parallel with presentation of content of a different application.

[00148] Method 2500 may receive multiple selections. In response to these multiple selections, block 2508 causes content of each selected application to be presented in the multi-application environment. In such a case, manager 134 may serially present content from each of the selected applications or at once. In one serial case, consider the above example but assume that application 2706 was selected at a prior iteration of block 2506 and, responsive to this selection, manager 134 presented content 2616 of application 2706 (as shown in Figs. 26 and 27). Following this selection of application 2706, later iterations of blocks 2506 and 2508 are performed. Serial presentation of applications is enabled by manager 134 at least through additional iterations of block 2506 and 2508, shown as potentially repeating with a dashed line from block 2508 to block 2506.

Desktop as an Immersive Interface in a Multi-Application Environment

[00149] Fig. 28 depicts a method 2800 enabling display of a windows-based desktop as an immersive interface within a multi-application environment. In portions of the following discussion reference may be made to system 100 of Fig. 1 and other methods and example embodiments described elsewhere herein, reference to which is made for example only.

[00150] Block 2802 displays a desktop as an immersive interface within a multi-application environment. The multi-application environment, like those described elsewhere herein, is configured to support access to multiple applications. Thus, a user may see content through an interface associated with an application (or multiple

applications if multiple applications include interfaces on the desktop) and interact with that application through the interface, all through the multi-application environment.

[00151] Block 2804 enables interaction with the desktop immersive interface. This interaction can be simultaneous with interactions enabled for other interfaces. Further, this interaction can be through the multi-application environment and also include windows-based and desktop-based controls, such as window-frame controls and a taskbar, respectively.

[00152] Windows-based mode module 122 and multi-application environment module 124, acting separately or in conjunction with operating system 120 of Fig. 1, for example, may treat a desktop immersive interface as one of applications 136 and windows and taskbars, etc., as various examples of application user interfaces 138.

[00153] By way of example, consider Fig. 29, which illustrates tablet computing device 106 having a desktop immersive interface 2902 displaying windows-based interfaces 2904 and taskbar 2906, and an immersive interface 2908 displaying content 2910, all within multi-application environment 2912. Desktop immersive interface 2902 may also include representations (e.g., icons) of applications executable within desktop immersive interface 2902 and representations of folders to support a hierarchical file structure of computing device 102, to name just a few.

[00154] Further, the functionality provided within the desktop immersive interface 2902 may behave differently than other interfaces within multi-application environment 2912, such as those common to windows-based interfaces.

[00155] Block 2806, responsive to interaction, provides the interaction to an application within the desktop immersive interface. In some cases this includes passing a gesture or gesture portion to an application having an interface within the desktop immersive interface, such as to operating system 120 for an interaction with taskbar 2906, or to an application associated with one of window-based interfaces 2904. In other cases this may include passing keystrokes to a primary (e.g., up-front) window of windows-based interfaces 2904. In so doing, the techniques permit users to engage with applications and interfaces common to desktop, windows-based environments, while also permitting interaction with immersive interfaces and other operations of a multi-application environment.

[00156] A desktop immersive interface can be managed in a manner similar or identical to those described elsewhere herein. An edge gesture, for example, may be used to select, move, or size a desktop immersive interface, such as desktop immersive interface 2902.

Further, a desktop immersive interface can behave as a single application as part of an application queue, and thus be selected or removed from a multi-application environment as noted for other interfaces above.

Multi-Application Environment

[00157] The techniques and apparatuses described above enable many different embodiments of a multi-application environment, including environments having one interface but permitting additional interfaces, multiple interfaces that are all immersive, multiple interfaces that are a mix of immersive and windows-based interfaces, and a desktop treated as an immersive interface. In some embodiments, these multi-application environments enable selection of various menus or additional interfaces for systems and applications offering additional controls. In still other embodiments, these multi-application environments enable gestures by which to manage applications and interfaces.

[00158] Fig. 30 depicts a method 3000 enabling content presentation and/or management of a multi-application environment. In portions of the following discussion reference may be made to system 100 of Fig. 1 and other methods and example embodiments described elsewhere herein, reference to which is made for example only.

[00159] Block 3002 presents multiple interfaces associated with multiple applications within a multi-application environment, at least one of the multiple interfaces being an immersive interface.

[00160] As noted elsewhere herein, the multi-application environment can present various combinations of different interfaces. Consider, for example, multi-application environments having at least one immersive interface as shown in each of Figs. 3, 7, 10, 12, 16, 18, 20, 23, 24, 26, 27, and 29. These example multi-application environments are described in order.

[00161] Fig. 3 illustrates a multi-application environment having a single immersive interface 304. Fig. 7 illustrates a multi-application environment having the single immersive interface of Fig. 3 along with application-selection interface 702. Fig. 10 illustrates a multi-application environment having windows-based email interface 1004 and two immersive interfaces 1006 and 1008. Fig. 12 illustrates the multi-application environment of Fig. 10 along with an interface enabling additional controls—email handling interface 1202. Fig. 16 illustrates a multi-application environment switching from webpage 306, responsive to a gesture selection, to media player 1604. Fig. 18 illustrates a multi-application environment having immersive interface 304 and thumbnail images 1602, 1808 of two other interfaces (one immersive and the other windows-based).

Fig. 20 illustrates a multi-application environment having two immersive interfaces, larger immersive interface 2006 and smaller immersive interface 2008 separated by immersive interface divider 2010. Fig. 23 illustrates a multi-application environment having current immersive interface 2302 and regions 2306 and 2310 in which other interfaces can be snapped to fully occupy. Fig. 24 illustrates the multi-application environment of Fig. 23 having second immersive interface 2404 showing a social-networking webpage 2406 for social-networking application 1510 of Fig. 15. Fig. 26 illustrates a multi-application environment presenting content through three immersive interfaces, one in primary region 2604 and the other two in non-primary region 2606. Fig. 27 illustrates the multi-application environment of Fig. 26 along with a user interface management menu 2700. Fig. 29 illustrates a multi-application environment 2912 having desktop immersive interface 2902 displaying windows-based interfaces 2904 and taskbar 2906, and an immersive interface 2908 displaying content 2910.

[00162] Block 3004 presents content through at least one of the multiple interfaces. Content is shown displayed in many figures described above, such as media being played, social-networking webpages, news website articles, and word-processing documents. The content being displayed is received from various sources, such as applications 136, which may have generated the content or received the content from a remote source (e.g., in cases of a web browser application receiving content from a remote provider).

[00163] Note that many of the mentioned figures show content presented concurrently. The multi-application environment can present moving, visual media, such as a movie, in one interface while at the same time presenting a webpage having a media slideshow in another interface, both in real time. Further, the multi-application environment enables interaction with multiple interfaces without necessarily requiring an initial selection. Thus, a user may select one interface or another, without having to first select an interface that is not primary or on top of a stack, as can be the case in windows-based environments.

[00164] Block 3006 enables selection to alter a size or a location in the multi-application environment or remove from the multiple-application environment at least one of the multiple interfaces. Block 3006 may act in various manners described elsewhere herein, such as to enable selection to move an interface from one region of the multi-application environment to another with a drag-and-drop gesture.

[00165] Block 3008, responsive to receiving the selection, alters the size of, alters the location of, or removes the selected interface from the multi-application environment. Example alterations to sizes and locations of interfaces in the multi-application

environment are shown in Figs. 23 and 24. Fig. 23 illustrates multi-application environment 2304 having at first current immersive interface 2302 and then having reduced-size immersive interface 2402 and with a second immersive interface 2404 showing a social-networking webpage 2406.

[00166] Block 3010, responsive to the same selection, alters a size or location of another interface in the multi-application environment. This is also shown in Figs. 23 and 24. While not shown, multi-application environment module 124 responds to an interface being removed by resizing and/or re-locating other interface(s) within the multi-application environment. Assume a case where multi-application environment 2304 includes both interfaces 2402 and 2404 of Fig. 24. Responsive to a selection to remove second immersive interface 2402, multi-application environment module 124 returns to multi-application environment 2304 as shown in Fig. 23, which includes only current immersive interface 2302. Note that current immersive interface 2302 is larger than, and occupies a region of, second immersive interface 2404.

[00167] The preceding discussion describes some methods in which the techniques manage immersive interfaces in a multi-application environment, some other methods that enable switching back to a previously-interacted-with application, still other methods that describe ways in which the techniques enable and/or use edge gestures, further methods describing ways in which the techniques enable and/or use a desktop as an immersive interface, and a method enabling content presentation and/or management of a multi-application environment. These methods are shown as sets of blocks that specify operations performed but are not necessarily limited to the order shown for performing the operations by the respective blocks. Furthermore, these methods, in whole or in part, can be used in combination.

[00168] Aspects of these methods may be implemented in hardware (*e.g.*, fixed logic circuitry), firmware, a System-on-Chip (SoC), software, manual processing, or any combination thereof. A software implementation represents program code that performs specified tasks when executed by a computer processor, such as software, applications, routines, programs, objects, components, data structures, procedures, modules, functions, and the like. The program code can be stored in one or more computer-readable memory devices, both local and/or remote to a computer processor. The methods may also be practiced in a distributed computing environment by multiple computing devices.

Example Device

[00169] Fig. 31 illustrates various components of example device 3100 that can be implemented as any type of client, server, and/or computing device as described with reference to the previous Figs. 1-31 to implement techniques enabling and using edge gestures, switching back to a previously-interacted with application, and/or managing an immersive interface in a multi-application environment. In embodiments, device 3100 can be implemented as one or a combination of a wired and/or wireless device, as a form of television client device (*e.g.*, television set-top box, digital video recorder (DVR), etc.), consumer device, computer device, server device, portable computer device, user device, communication device, video processing and/or rendering device, appliance device, gaming device, electronic device, and/or as another type of device. Device 3100 may also be associated with a user (*e.g.,* a person) and/or an entity that operates the device such that a device describes logical devices that include users, software, firmware, and/or a combination of devices.

[00170] Device 3100 includes communication devices 3102 that enable wired and/or wireless communication of device data 3104 (*e.g.*, received data, data that is being received, data scheduled for broadcast, data packets of the data, etc.). The device data 3104 or other device content can include configuration settings of the device, media content stored on the device, and/or information associated with a user of the device. Media content stored on device 3100 can include any type of audio, video, and/or image data. Device 3100 includes one or more data inputs 3106 via which any type of data, media content, and/or inputs can be received, such as user-selectable inputs, messages, music, television media content, recorded video content, and any other type of audio, video, and/or image data received from any content and/or data source.

[00171] Device 3100 also includes communication interfaces 3108, which can be implemented as any one or more of a serial and/or parallel interface, a wireless interface, any type of network interface, a modem, and as any other type of communication interface. The communication interfaces 3108 provide a connection and/or communication links between device 3100 and a communication network by which other electronic, computing, and communication devices communicate data with device 3100.

[00172] Device 3100 includes one or more processors 3110 (*e.g.*, any of microprocessors, controllers, and the like), which process various computer-executable instructions to control the operation of device 3100 and to enable techniques enabling a multi-application environment. Alternatively or in addition, device 3100 can be implemented with any one

or combination of hardware, firmware, or fixed logic circuitry that is implemented in connection with processing and control circuits which are generally identified at 3112. Although not shown, device 3100 can include a system bus or data transfer system that couples the various components within the device. A system bus can include any one or combination of different bus structures, such as a memory bus or memory controller, a peripheral bus, a universal serial bus, and/or a processor or local bus that utilizes any of a variety of bus architectures.

[00173] Device 3100 also includes computer-readable storage media 3114, such as one or more memory devices that enable persistent and/or non-transitory data storage (*i.e.*, in contrast to mere signal transmission), examples of which include random access memory (RAM), non-volatile memory (*e.g.*, any one or more of a read-only memory (ROM), flash memory, EPROM, EEPROM, etc.), and a disk storage device. A disk storage device may be implemented as any type of magnetic or optical storage device, such as a hard disk drive, a recordable and/or rewriteable compact disc (CD), any type of a digital versatile disc (DVD), and the like. Device 3100 can also include a mass storage media device 3116.

[00174] Computer-readable storage media 3114 provides data storage mechanisms to store the device data 3104, as well as various device applications 3118 and any other types of information and/or data related to operational aspects of device 3100. For example, an operating system 3120 can be maintained as a computer application with the computer-readable storage media 3114 and executed on processors 3110. The device applications 3118 may include a device manager, such as any form of a control application, software application, signal-processing and control module, code that is native to a particular device, a hardware abstraction layer for a particular device, and so on.

[00175] The device applications 3118 also include any system components or modules to implement the techniques, such as device applications 3118 including multi-application environment module 124, system-interface module 126, gesture handler 128, application manager 130, manager 134, and application(s) 136.

Conclusion

[00176] Although embodiments of techniques and apparatuses enabling a multi-application environment have been described in language specific to features and/or methods, it is to be understood that the subject of the appended claims is not necessarily limited to the specific features or methods described. Rather, the specific features and methods are disclosed as example implementations enabling a multi-application environment.

CLAIMS

1. A computer-implemented method comprising:

presenting multiple interfaces associated with multiple applications within a multi-application environment, at least one of the multiple interfaces being an immersive interface; and

presenting content through at least one of the multiple interfaces.

2. A computer-implemented method as described in claim 1, further comprising presenting second content concurrently with presenting the first-mentioned content, the second content presented through another of the multiple interfaces.

3. A computer-implemented method as described in claim 2, wherein presenting the first-mentioned content presents first moving, visual media in real time and presenting the second content presents second moving, visual media in real time.

4. A computer-implemented method as described in claim 1, wherein the immersive interface represents a desktop, the desktop having at least one windows-based interface or a taskbar.

5. A computer-implemented method as described in claim 4, further comprising enabling interaction with an application associated with the windows-based interface and through the multi-application environment.

6. A computer-implemented method as described in claim 4, wherein the desktop further includes a task bar, and further comprising enabling interaction, through the multi-application environment, with the task bar.

7. A computer-implemented method as described in claim 1, wherein the content is received by the multi-application environment from one of the multiple applications.

8. A computer-implemented method as described in claim 1, wherein another of the multiple interfaces is a windows-based interface.

9. A computer-implemented method as described in claim 1, wherein the multi-application environment does not include a visible control.

10. A computer-implemented method as described in claim 1, further comprising:

enabling selection to alter a size or a location in the multi-application environment or remove from the multiple-application environment at least one of the multiple interfaces; and

responsive to receiving the selection, altering the size of, altering the location of, or removing the selected interface from the multi-application environment.

presenting content through two or more of the multiple interfaces and at a same time;

enabling selection to alter a size or a location of one of the multiple interfaces; and

responsive to receiving the selection, concurrently altering the size or the location of the one of the multiple interfaces and another of the multiple interfaces.