

Petitioner Masimo Corporation's Demonstratives For Oral Hearing

August 1, 2024

Apple Inc. v. Masimo Corporation
IPR2023-00664

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)

See also claims 9 and 17

2. Vary navigational paths (Dep. Claims)

a. Return to wake screen with different swipe gesture (Claim 6)

See also claims 14 and 22

b. Use consistent swipe gesture for home screen (Claim 3)

See also claims 11 and 19

c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)

See also claims 12 and 20

d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)

See also claims 13 and 21

e. Navigate to control panel (Claims 7, 8)

See also claims 15, 16, 22, and 23

3. Add limitations of substitute claims (Sub. Claims)

Claim 1

1. A method comprising:
at computer system that is in communication with a display generation component and one or more input devices:
while the computer system is in a power saving state, detecting an input that meets display-waking criteria;
in response to detecting the input that meets the display-waking criteria, displaying, via the display generating component, a wake screen user interface;
while displaying the wake screen user interface, detecting a first input that is directed to a portion of the wake screen user interface and includes first movement; and
in response to detecting the first input that is directed to the portion of the wake screen user interface:
in accordance with a determination that the first input meets first criteria, wherein the first criteria require the first movement to be in a first direction in order for the first criteria to be met:
displaying of a home screen user interface that is different from the wake screen user interface,

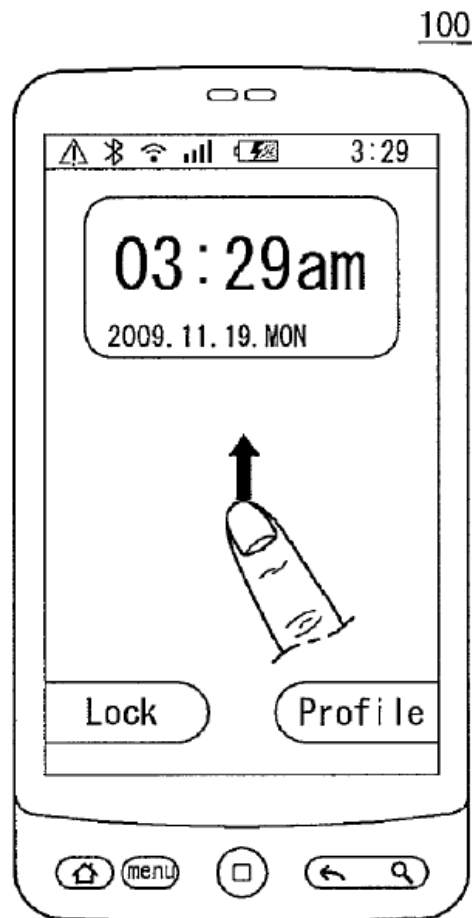
wherein the home screen user interface includes a plurality of application icons corresponding to different applications, and wherein a respective application icon of the plurality of application icons, when selected, causes display of an application corresponding to the respective application icon; and
in accordance with a determination that the first input meets second criteria different from the first criteria, wherein the second criteria require the first movement to be in a second direction that is different from the first direction in order for the second criteria to be met:
displaying a widget screen user interface that is different from the wake screen user interface and the home screen user interface, wherein the widget screen user interface includes a plurality of user interface objects corresponding to different applications, wherein a respective user interface object of the plurality of user interface objects contains application content from an application corresponding to the respective user interface object, and when selected, causes display of an application corresponding to the respective user interface object.

Claim 1 is unpatentable over Chae alone or with Narendra

1. Claim 1

Chae discloses the preamble and every limitation of claim 1, and, thus, anticipates or renders obvious claim 1. Moreover, if Chae is interpreted as lacking any limitation of claim 1, it would have been obvious to add such limitation to Chae in view of Narendra.

Chae discloses computer connected to display and input

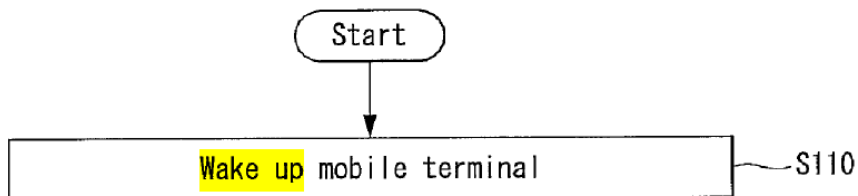


- Chae's smart phone or laptop computer is a "computer system"
- Chae's touch screen is a "display" (e.g., it displays a clock in Figure 4(a))
- Chae's touch screen is also an "input device"
- Computer system necessarily communicates with display and input device

Chae discloses a computer in a power-saving state

Chae:

FIG. 2



Dr. Rosenberg (EX1003):

78. A POSITA would have understood Chae's disclosure that the mobile terminal "wakes up" to refer to the typical wake-up process that was well-known and commonly implemented in smartphones and similar mobile devices. Specifically, a POSITA would have understood that the mobile terminal was in "a power saving state," with its display darkened, prior to waking up, and that the display turns on upon waking up. A POSITA would have understood that the "user's action such as clicking a button" referenced in Chae Paragraph 94 is an "input that meets display-waking criteria" because the user's action does, in fact, cause the mobile terminal (including its display) to wake up.

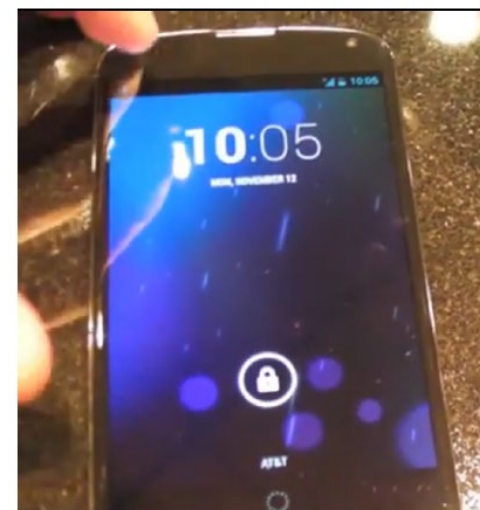
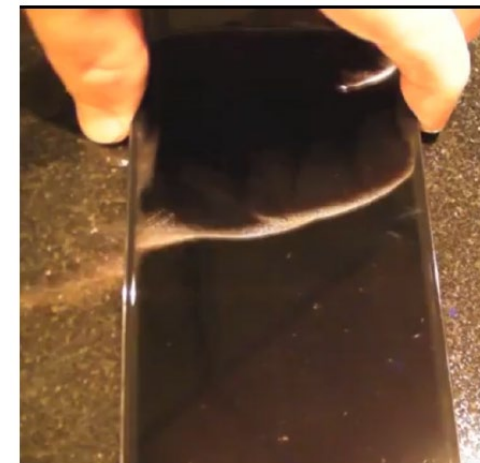
EX1019:

FIG. 5 shows an example state machine that can be used when finger print matching is desired. The state machine can be executed in the controller 330, the application processor 350 or a combination thereof. In state 510, the display is locked and in a sleep state. In the sleep state, the touch panel display is not illuminated and any sensor functionality is disabled. As a result, a low-power mode is achieved. When

Dr. Rosenberg (EX1056):

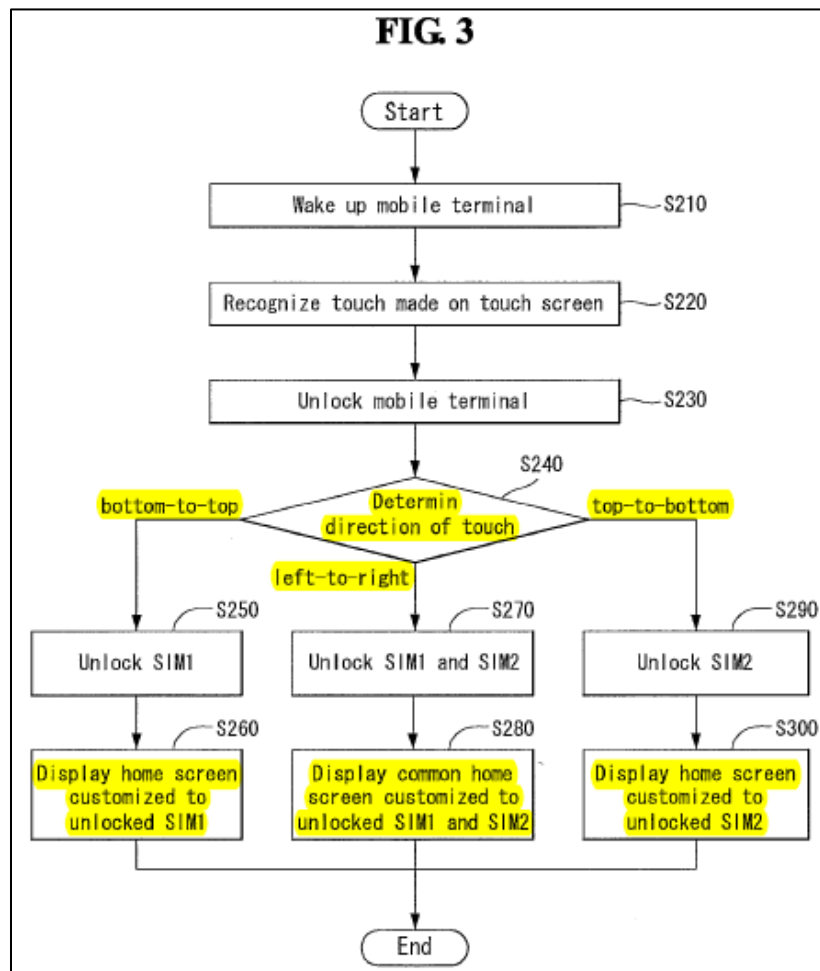
26. Apple argues that Chae and Android do not expressly state that their devices are in a power-saving state when their displays are off. POR at 19-20. However, it is undisputable that a display consumes less power when it is off. I understand that Dr. Cockburn conceded that a display consumes less power when it is off. EX1057, 95:17-96:3. Dr. Cockburn and Apple speculate that Chae might not save power with its display off because its other components may consume more power. *Id.*; POR at 20. However, nothing in Chae supports that speculation. In my view, a POSITA would have understood Chae as teaching that the computer system uses less power when its display is off than when its display is on.

Android:

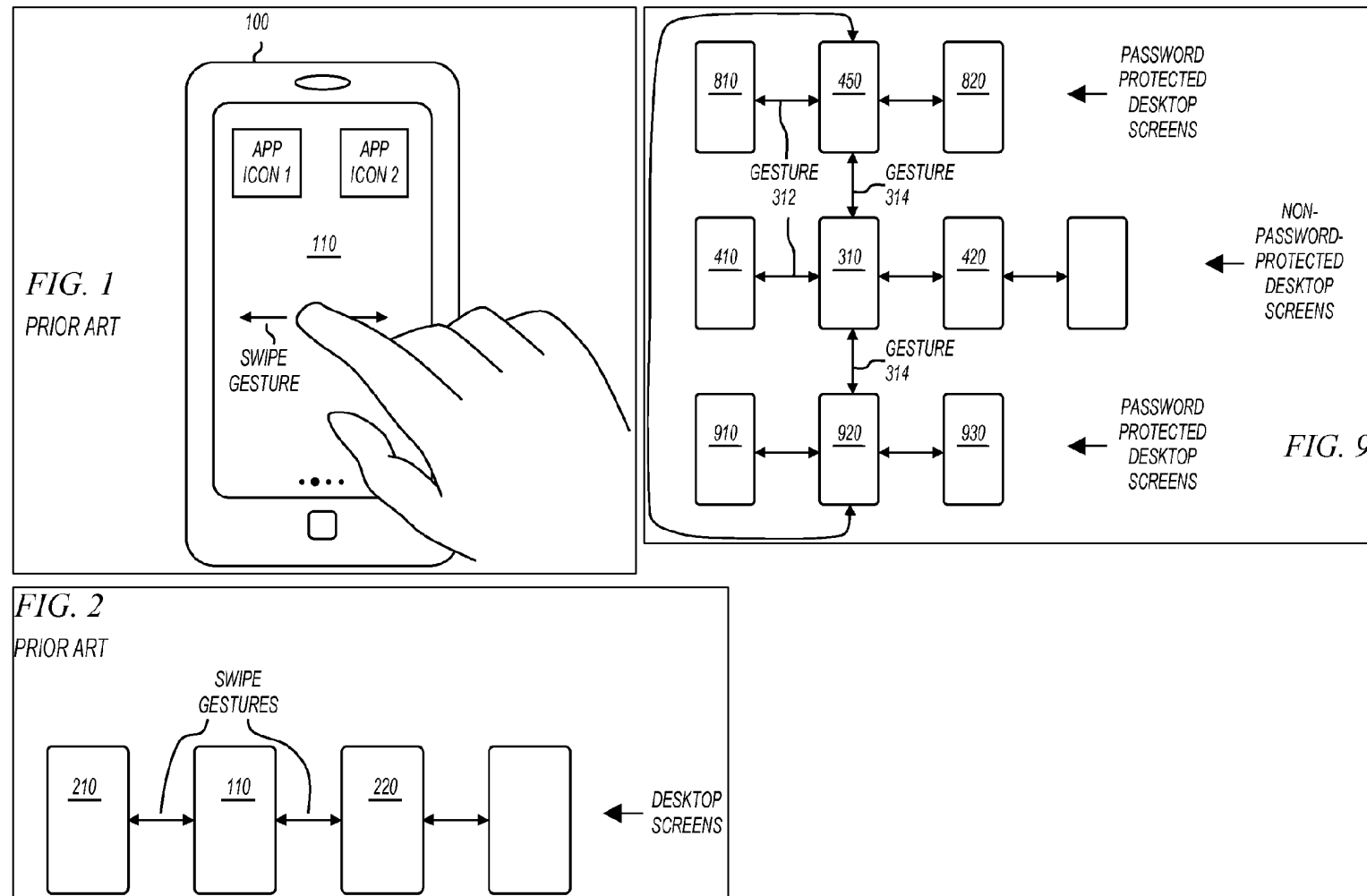


Chae and Narendra disclose swiping for screen navigation

Chae:



Narendra:



Chae discloses a wake screen user interface

'352 Patent:

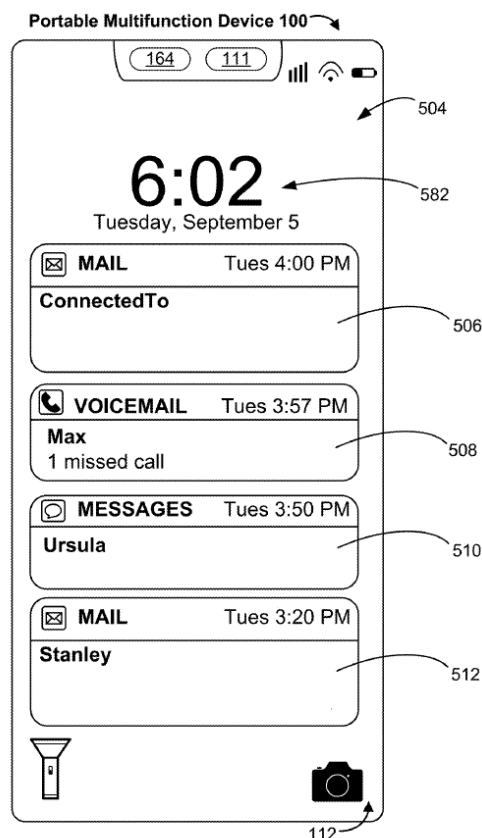
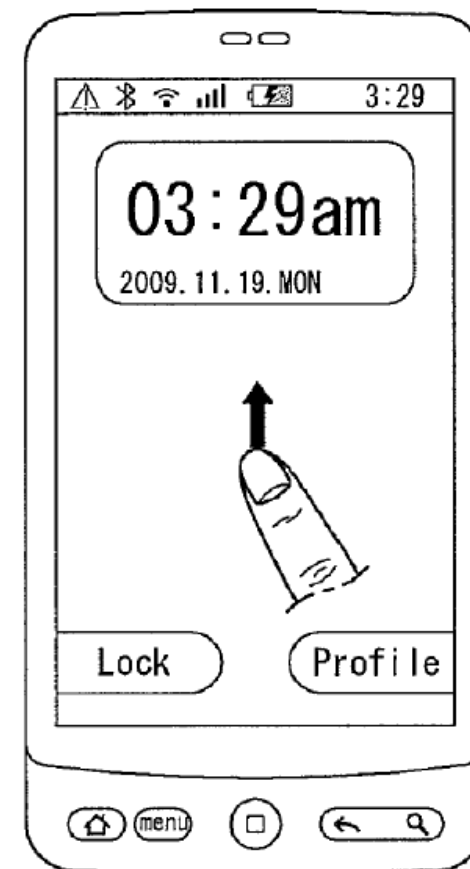
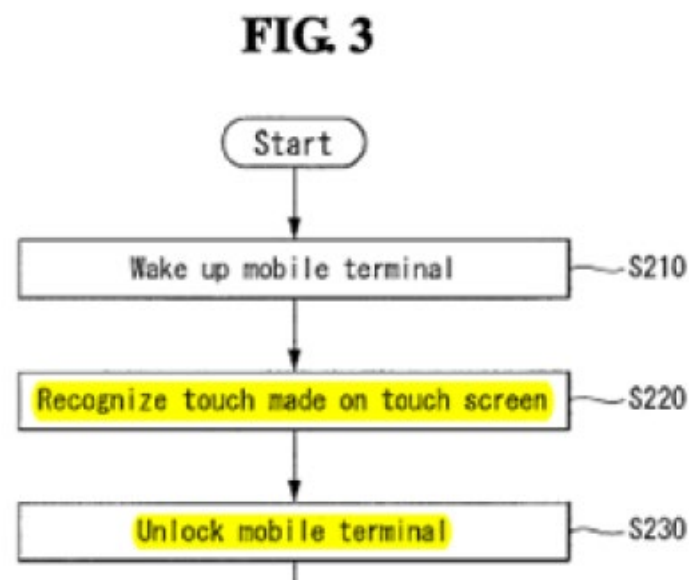


Figure 5B

- Displayed upon waking the device
- No required content or features on wake screen

Chae:

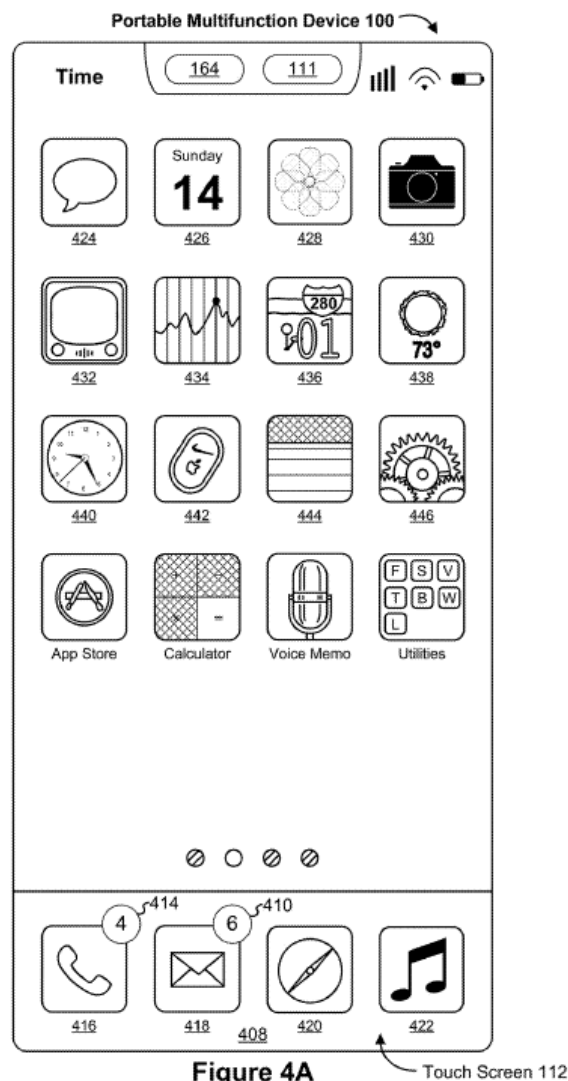


(a)

Chae discloses a home screen user interface

'352 Patent:

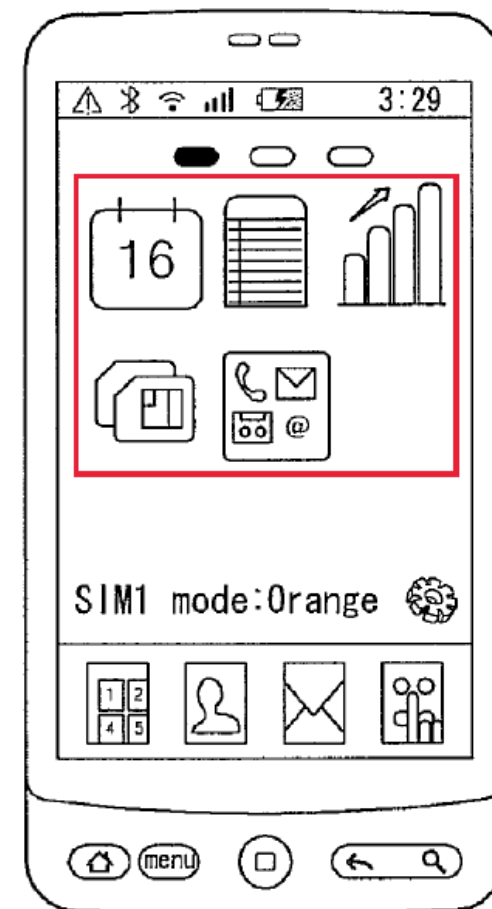
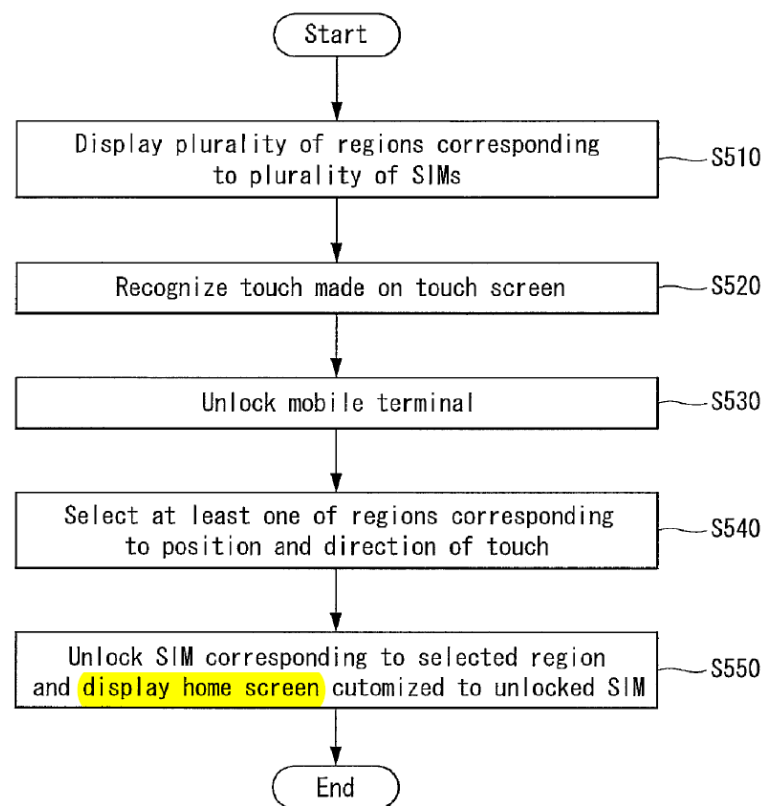
- Includes multiple application icons



Touch Screen 112

Chae:

FIG. 17

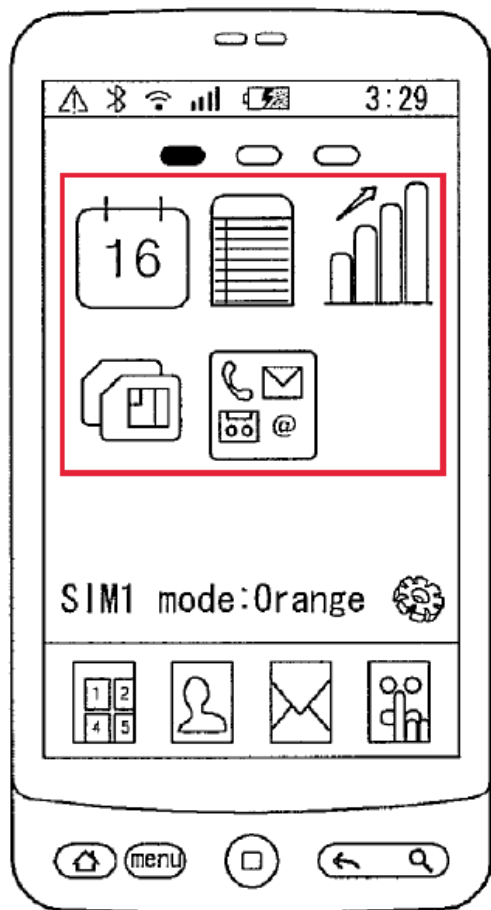


(b)

If Chae did not disclose opening applications, Narendra does

Chae:

100



(b)

Narendra:

Application icons that display on desktop screens are used to launch applications. For example, a user might launch a music player application by tapping on icon 330. Also for example, the user might launch an email application by tapping on icon 332. Both application icons 330 and 332 are displayed on desktop screen 310.

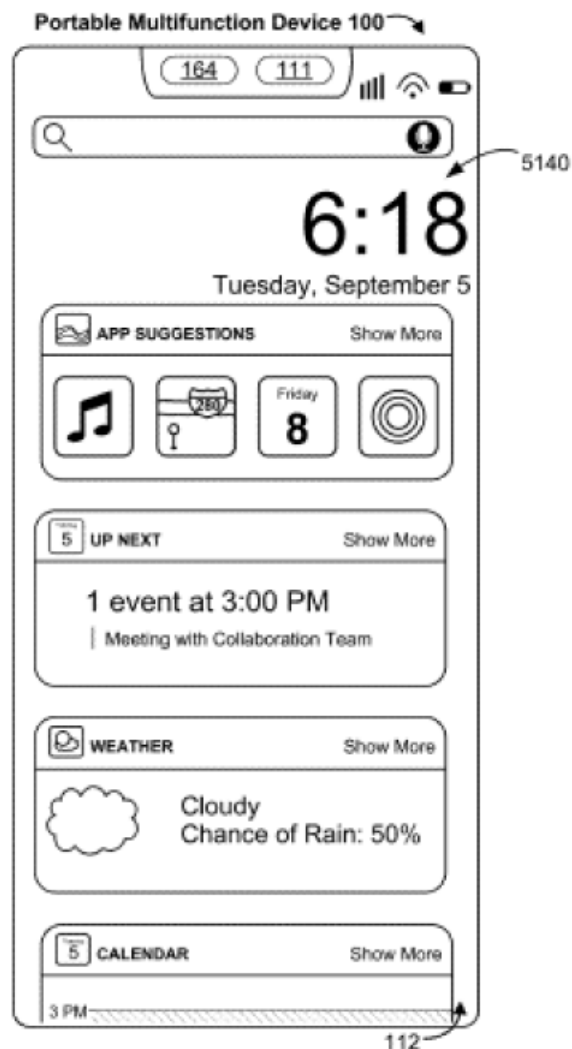
Dr. Rosenberg:

86. Accordingly, Chae discloses limitation 1f. If Chae were somehow interpreted as *not* disclosing that its “icons” are “corresponding to different applications, and wherein a respective application icon of the plurality of application icons, when selected, causes display of an application corresponding to the respective application icon,” this functionality would have been obvious in view of Narendra. A POSITA would have been motivated by basic GUI design principles and by the teachings of Narendra to add Narendra’s basic GUI functionality to Chae so that the graphical user interface of Chae would have the expected functionality of launching a corresponding application when a user selects a given icon.

Chae discloses a widget screen user interface

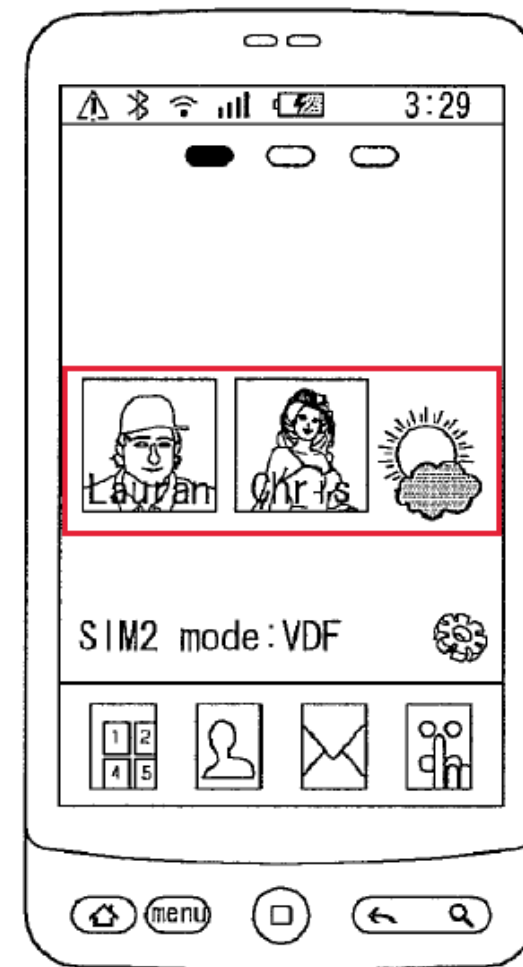
'352 Patent:

- Includes multiple “widgets” (“user interface objects containing application content from an application . . . and when selected, causes display of an application”) (claim 1)



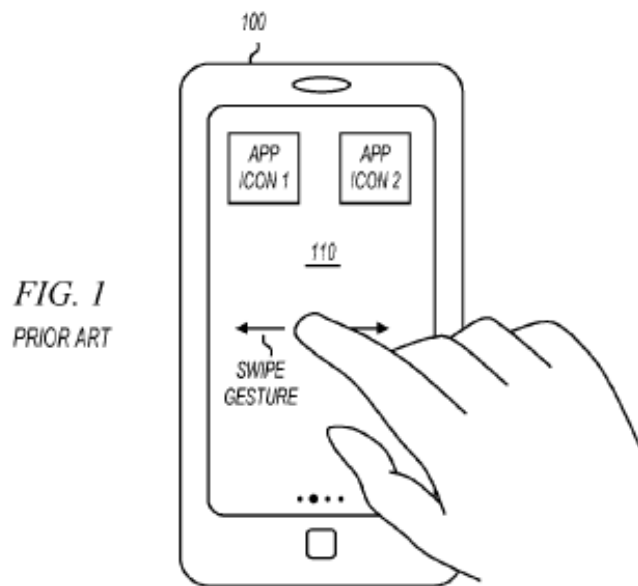
Chae:

- “Here, the home screen customized to the SIM has items includ[ing] icons and **widgets**.” EX1005 ¶ 100

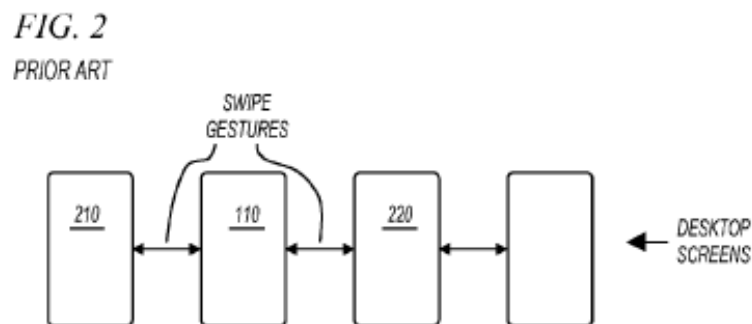


(b)

Narendra discloses a widget screen user interface



“Desktop screens on mobile device 300 display content such as application icons, **widgets**, and the like.” EX1006 ¶ 25



Hong discloses widgets

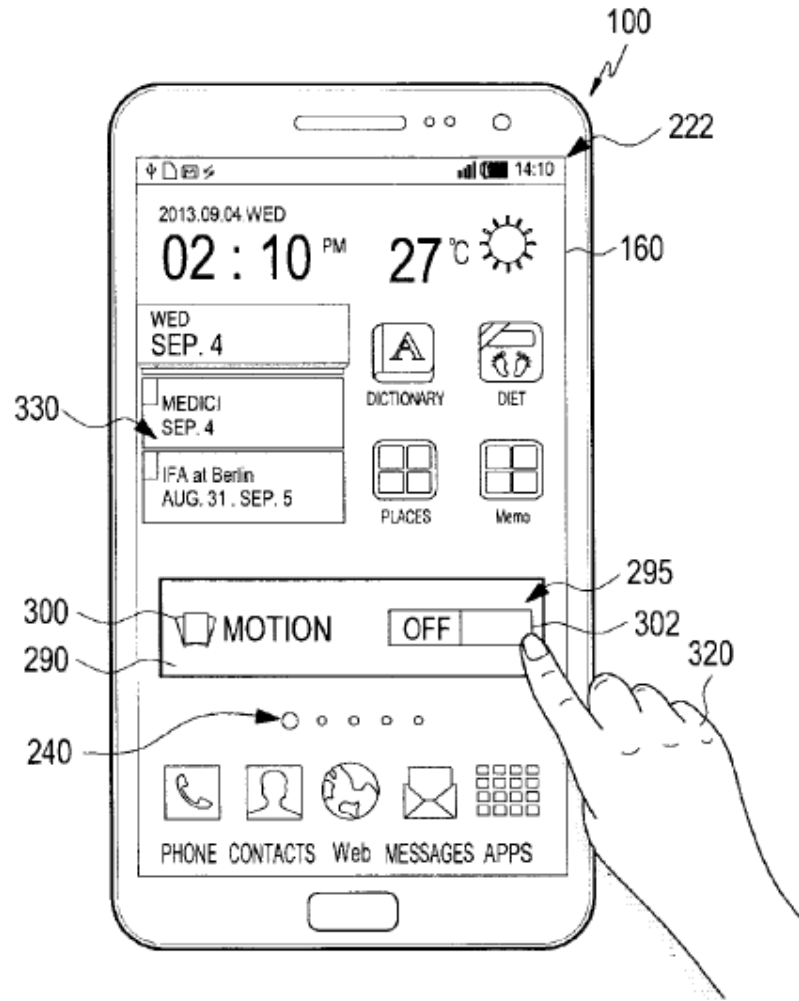


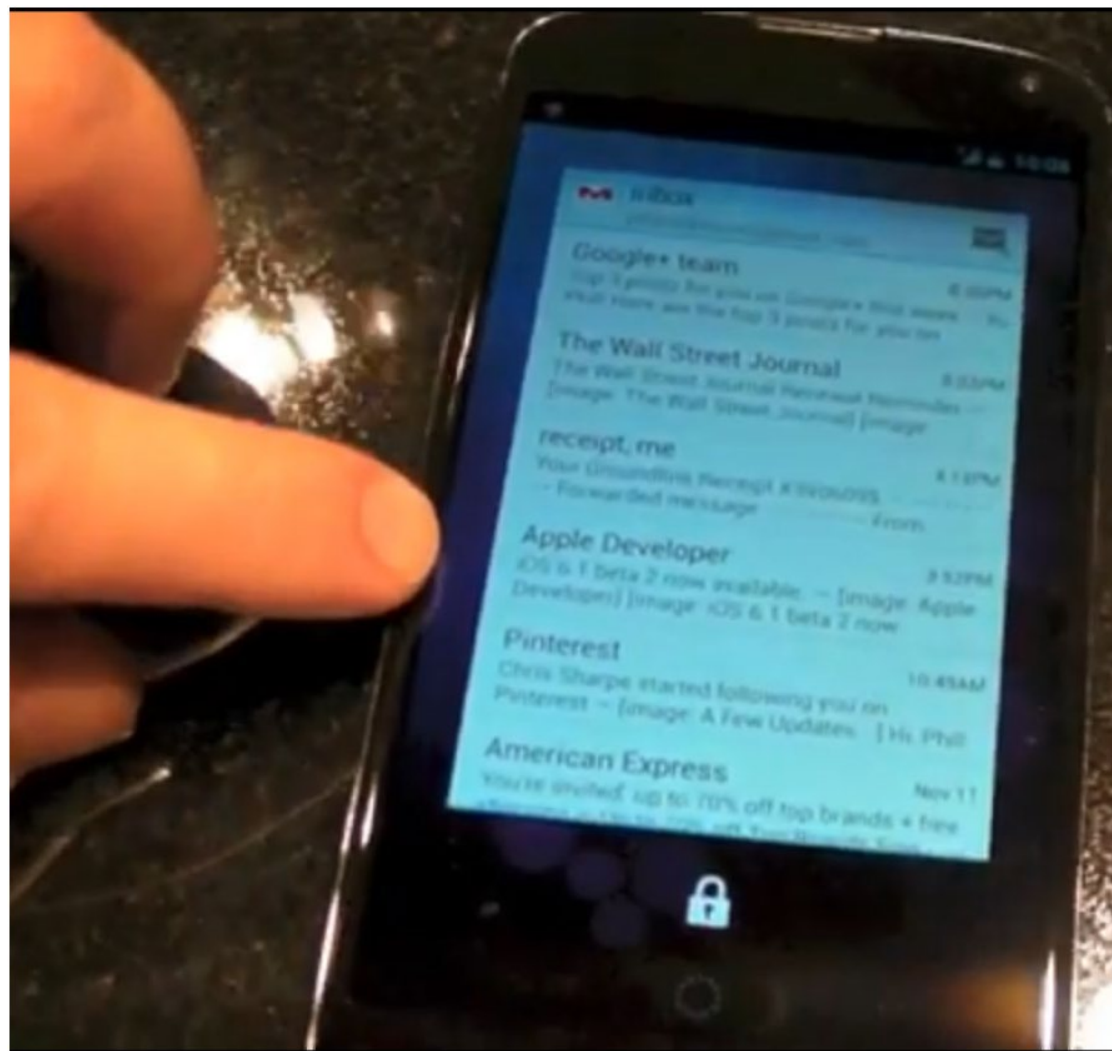
FIG. 7A

“Recently, application **widgets** have been provided that **enable a user to access contents provided in an application without executing the application, such as a clock, a calendar**”

“A user may check ... information ... provided by an application ... as a **preview** ... without actually executing the application itself.”

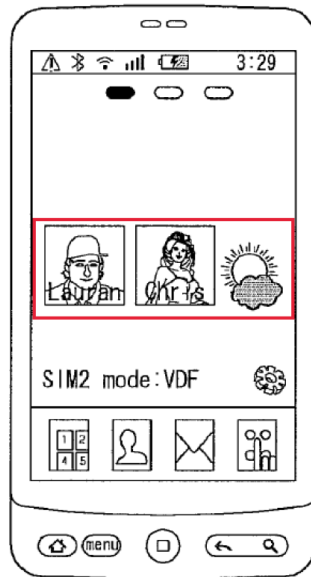
EX1010 ¶¶ 4-5

Android discloses widgets



Chae's Figure 6(b) discloses a widget screen

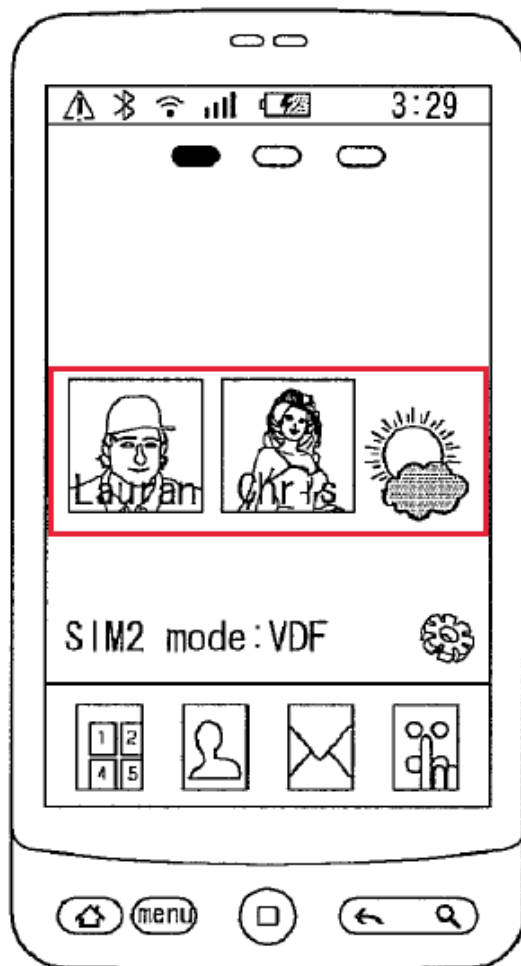
89. Finally, a POSITA would have understood that two or more of the user interface objects shown on Figure 6(b) are “user interface objects” (widgets) which “contain[] application content from an application corresponding to the respective user interface object.” For example, a POSITA would have understood that the three objects shown in the red box of the annotated Figure 6(b) below are “user interface objects” as claimed:



(b)

Id., Fig. 6(b) (annotated). Specifically, the object on the left and in the center include images and names of people (Lauran and Chris) and the object on the right includes a graphical depiction of current weather conditions (partly cloudy). A POSITA would have understood that this information is “application content from an application corresponding to the respective user interface object.” For example, a POSITA would have understood that (1) displaying the images and names of Lauran and Chris would involve retrieving that content from a corresponding application, such as an image-editing application or an application for managing contacts or profiles and (2) displaying a graphical depiction of current weather conditions would involve retrieving information indicating those current weather conditions from a weather application.

Chae's Figure 6(b) discloses a widget screen



(b)

91. Similar to icons, widgets were common and well-known elements of GUIs before both the filing of the '352 patent and Chae's publication. A POSITA would have known that part of the well-known functionality of widgets was that selecting a widget would cause "display of an application corresponding to the respective user interface object." Figure 6(b), in view of a POSITA's knowledge of this well-known widget functionality, would have conveyed to a POSITA that selecting one of the objects shown in the center of the screen (such as the weather object) or one of the objects on the status bar (such as the digital clock) would cause display of a corresponding application (such as a weather application or a digital clock application). Therefore, a POSITA would have understood that the screen shown in Figure 6(b) is a "widget screen user interface."

Hong supports Dr. Rosenberg's understanding of widgets

93. A POSITA would have understood, in view of Chae's and Narendra's disclosures, that those references used the term "widget" to refer to a user interface object that displays content generated by an associated application without needing to (but being able to) open the application. Hong, an application filed in 2014 and published in 2015, supports this understanding. Specifically, Hong's background section, which describes prior art, states:

The portable phone provides various applications. Recently, application widgets have been provided that enable a user to access contents provided in an application without executing the application, such as a clock, a calendar, a memo, search, a map, news, a real-time camera, and the like.

As noted above, application widgets have been provided that enable a user to access contents in an application without executing the application. A user may check, at a glance, information or a function provided by an application through a widget that functions as a preview of the information or the function without actually executing the application itself.

EX1010 ¶¶ 4-5. Figure 7A depicts several such widgets on a screen:

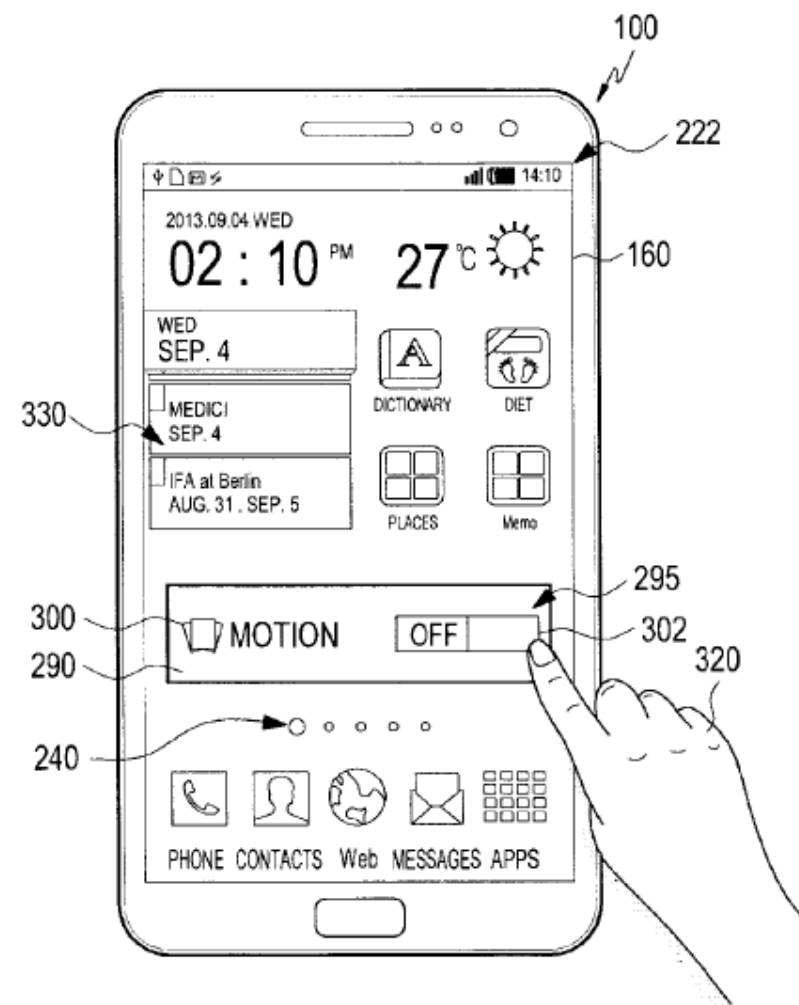


FIG. 7A

Apple argues that prior art widgets did not allow execution of applications

32. Moreover, both Hong and Android disclose or suggest displaying a corresponding application upon selection of the widget. Hong expressly discloses:

Recently, application widgets have been provided that enable a user to access contents provided in an application without executing the application, such as a clock, a calendar, a memo, search, a map, news, a real-time camera, and the like.

EX1010 ¶4. Hong further discloses that content displayed by a widget “functions as a preview of the information or the function without actually executing the application itself.” *Id.* ¶5. Apple interprets this disclosure to mean that widgets do not even allow execution of the corresponding application. In my view, Apple’s interpretation is implausible, for at least four reasons. First, Apple reads disclosure into Hong that is not there. Hong never says the user cannot execute the application. Second, Hong’s disclosure that a widget displays a “preview” of application content “without actually executing the application itself” suggests that selecting the widget will execute the application to display more content. Third, a POSITA would have understood that the widget applications Hong lists, such as a clock, calendar, map, or news, are the type of applications that can be executed to display more content. Fourth, a POSITA would have read Hong consistently with the basic GUI functionality that selecting an application icon launches an application.

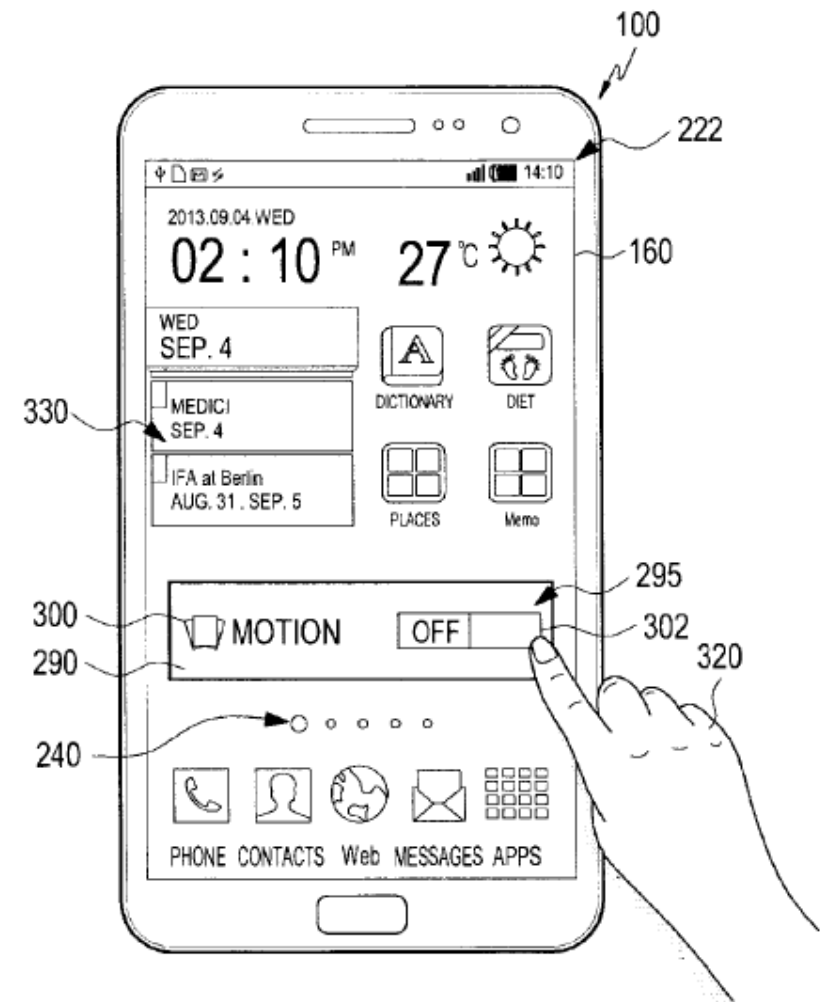


FIG. 7A

Response to Sur-reply Hong Argument

Sur-reply at 13 (citing EX1010 ¶ 68):

Specifically, Hong discloses that the “preview application screen displayed on display unit 160 via the widget may be identical to the application screen shown when the user actually executes the application” and the widget can display “the entirety of an application screen.” EX1010, [0068]-[0069], [0088]. Hong also

EX1010 ¶ 68:

[0068] Step S140 corresponds to a step of displaying a modified widget, in which the controller 170 displays a widget that provides a preview of the application screen on the display unit 160 without executing the selected application. The preview application screen displayed on display unit 160 via the widget may be identical to the application screen shown when the user actually executes the application. The widget may display a preview of the application screen that would be shown first, when the user actually executes the application. The controller 170 may detect a user selection of

EX1010 ¶¶ 4-5:

BACKGROUND

[0003] Generally, portable phones have been widely utilized due to their mobility. A portable phone supports various user functions in addition to mobile communication and a user operates the various functions using the portable phone.

[0004] The portable phone provides various applications. Recently, application widgets have been provided that enable a user to access contents provided in an application without executing the application, such as a clock, a calendar, a memo, search, a map, news, a real-time camera, and the like.

[0005] As noted above, application widgets have been provided that enable a user to access contents in an application without executing the application. A user may check, at a glance, information or a function provided by an application through a widget that functions as a preview of the information or the function without actually executing the application itself.

Apple ignores Android's widget opening an email application

With respect to Android, Apple ignores a portion of the video showing a device displaying an email application in response to the user selecting an email widget. EX1007, 2:08-2:14. Thus, Apple is incorrect that Masimo relies solely on Rosenberg's opinion to show that selecting a widget to display a corresponding application was "well-known functionality." POR, 23-24. Hong and Android corroborate Rosenberg's opinion. Further, neither Apple nor Cockburn affirmatively deny that selecting a widget to display a corresponding application was well-known functionality. POR, 23-24; APPLE-2003 ¶48. EX1056 ¶33.

If Chae lacked widgets, they would have been obvious to add

96. Even if Chae's customized Figure 6(b) home screen did not include widgets—and, again, it is my opinion that Figure 6(b) shows at least three widgets—a POSITA would have been motivated to add multiple widgets to Chae's customized Figure 6(b) home screen, thereby making it a “widget screen user interface” as claimed. It was well known at the time of Chae's publication and before the effective

filing date of the '352 patent, that widgets advantageously display useful information generated by applications running in the background, without requiring the user to open the applications. For example, common widgets displayed the date and time of day, weather information such as the local temperature, and financial information such as stock quotes, without requiring the user to open a clock application, a calendar application, a weather application, or a financial application. The efficiency and convenience of using widgets to view such useful information without needing to open applications would have motivated a POSITA to include multiple widgets on Chau's customized Figure 6(b) home screen. A POSITA would have been motivated specifically to include *multiple* widgets (not just one) on a single screen so that as much useful information as possible could be viewed on a single screen without needing to switch between screens. Including multiple widgets on a single screen was a well-known technique of user-interface design to improve the user experience by reducing switching between screens. As mentioned above, this is explicitly disclosed in Figure 6(b) of Chae.

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)

See also claims 9 and 17

2. Vary navigational paths (Dep. Claims)

a. Return to wake screen with different swipe gesture (Claim 6)

See also claims 14 and 22

b. Use consistent swipe gesture for home screen (Claim 3)

See also claims 11 and 19

c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)

See also claims 12 and 20

d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)

See also claims 13 and 21

e. Navigate to control panel (Claims 7, 8)

See also claims 15, 16, 22, and 23

3. Add limitations of substitute claims (Sub. Claims)

Dependent claims – obvious variants of navigational paths

Claims 3-6 are dependent claims that merely recite different navigation paths for moving among the same wake, home, and widget screens recited in claim 1 in response to a swipe gesture in a particular direction.⁷ A POSITA would have understood these claims' added limitations are trivial and obvious variations of the basic concept of navigating from one user-interface screen to another using directional swipe gestures that both Chae and Narendra disclose. In view of Chae's and Narendra's disclosures, a POSITA would have recognized that directional swipe gestures are an efficient and convenient mechanism for accessing any user-interface screen from any other screen. EX1003 ¶104.

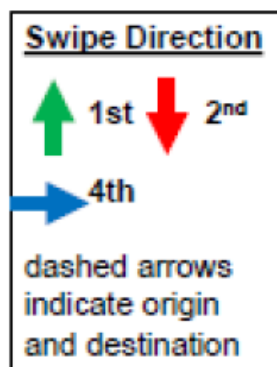
Dependent claims – obvious to choose from finite choices

Once a POSITA understood the basic prior-art concept of navigating between screens using directional swipe gestures, the POSITA would have understood that choosing a particular navigational path for moving among the user-interface screens and associating a particular directional swipe gesture with each movement is a simple matter of choosing from a finite number of design choices. For example, for each navigational path (such as from the wake screen to the home screen, from the home screen to the widget screen, etc.) the POSITA would simply need to choose from a finite set of directional swipe gestures (such as swipe left, swipe right, swipe up, swipe down, etc.) to associate with that navigational path. EX1003 ¶105.

Dependent claims – obvious to choose from finite choices

Accordingly, it would have been obvious to choose among the finite design choices described above to yield any of the specific navigational paths and associated directional swipe gestures recited by claims 3-6. EX1003 ¶107.

Conceptual graphical representations of claimed paths



⁸ The illustrative graphics for claims 3-8 focus on illustrating available navigational paths between various user-interface screens and do not illustrate other claim limitations, which are addressed elsewhere in this Petition. Further, the graphics illustrate *general* concepts, not a *precise* navigational map required by the claims. For example, the solid arrows in the graphic above are meant to illustrate that the first, second, and fourth directions are *different* from each other, *not* that they must be a bottom-to-top swipe, a top-to-bottom swipe, and a left-to-right swipe.

Alleged intended purposes would not prevent combination

16. Apple argues that combining Chae and Narendra would render Chae unsatisfactory for its intended purpose of using swipe gestures to navigate between user-interface screens associated with multiple SIM cards. POR at 10-13. I disagree for multiple reasons. First, the combination I proposed does not depend on Chae's specific teaching of using multiple SIM cards or Narendra's specific teaching of switching between password-protected and non-password-protected screens. I proposed using Chae's and Narendra's general concepts for navigating between user-interface screens using swipe gestures. Those general concepts have obvious uses beyond discriminating between screens with different SIM cards or levels of password protection. Dr. Cockburn admitted that those general concepts were well-

known, and that Chae and Narendra teach them. EX1057, 44:11-15, 46:12-48:3, 61:19-62:14. Dr. Cockburn also agreed that a two-reference combination need not incorporate every teaching from both references. *Id.*, 83:9-13, 84:8-14, 85:2-19. A POSITA would have found it obvious to combine Chae's and Narendra's general concepts without requiring fulfillment of Chae's purpose of switching between SIM cards or Narendra's purpose of switching between password-protected and non-password-protected screens. Further, as I explained in my first declaration, in my opinion, it would have been obvious to implement navigation between screens using swipe gestures on either a multi-SIM-card device like Chae or a single-SIM-card device like Narendra because the concepts disclosed in Chae are equally useful for devices with one SIM card or multiple SIM cards.

Alleged intended purposes would not prevent combination

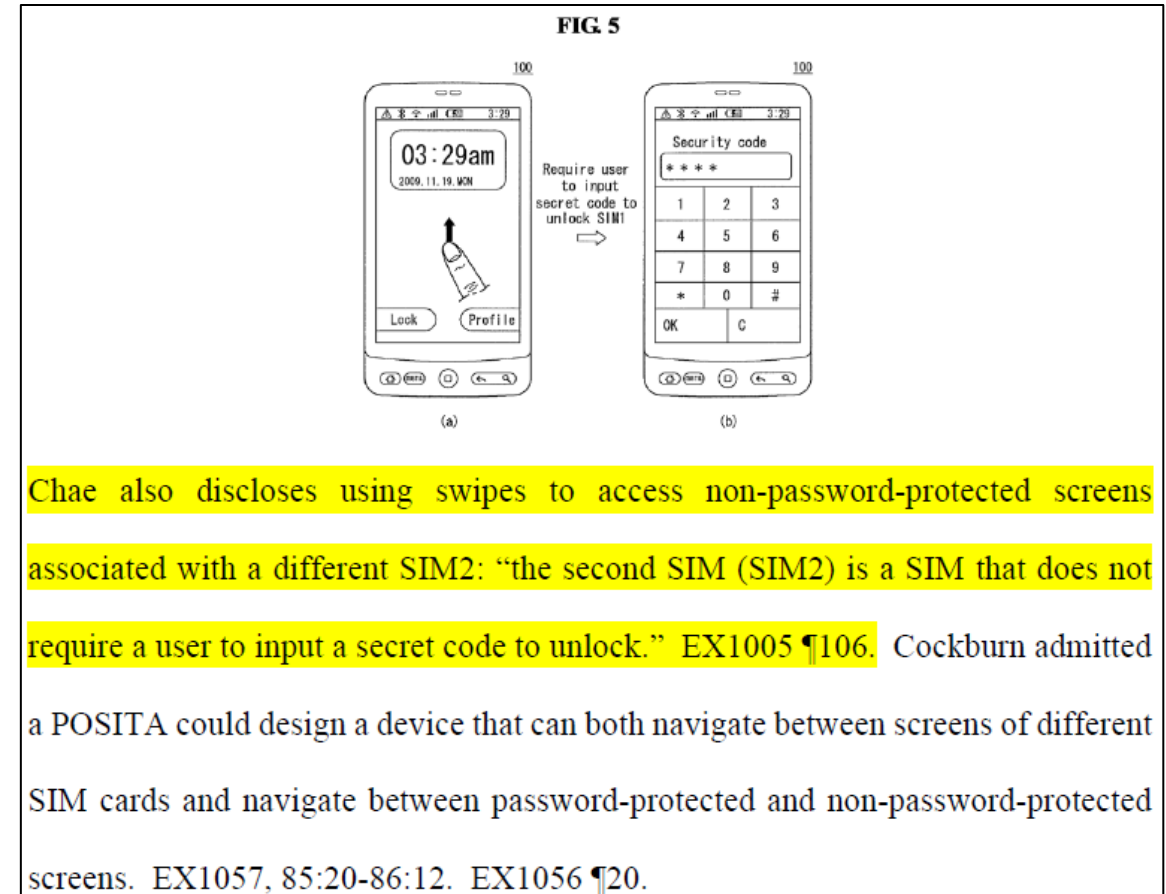
- “[F]amiliar items may have obvious uses beyond their primary purposes.” *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 420 (2007).
- A reference’s intended purpose does not control the obviousness inquiry. *Intel Corp. v. Qualcomm Inc.*, 21 F.4th 784, 800 (Fed. Cir. 2021).
- POSITA may be motivated to combine certain features to achieve claimed invention “even when bodily incorporation” of all disclosed but unclaimed features “would be impossible or inadvisable.” *Axonics, Inc. v. Medtronic, Inc.*, 73 F.4th 950, 957 (Fed. Cir. 2023).
- Prior-art patents are not limited to their described invention or problems solved; they are “relevant for all they contain.” *In re Heck*, 699 F.2d 1331, 1333 (Fed. Cir. 1983).

Narendra is compatible with Chae's multiple SIM cards

Second, Narendra's teachings are compatible with Chae's multi-SIM-card architecture. Apple does not dispute Chae and Narendra both teach the well-known concept of navigating between screens using swipes. EX1057, 44:11-15, 46:12-48:3, 61:19-62:14. Narendra more specifically teaches navigating in two dimensions using left and right swipes to move horizontally and up and down swipes to move vertically. *Id.*, 62:4-14, 75:9-13. A POSITA would have easily implemented the Narendra navigational concepts on various hardware platforms, including Chae's multi-SIM-card device. Such implementation on Chae's multi-SIM-card device would advantageously allow direct navigation from one screen associated with a SIM card to another screen associated with a different SIM card. EX1056 ¶17.

Chae is compatible with Narendra's password protection

Third, Apple fails to establish navigating between password-protected and non-password-protected screens is incompatible with navigating between screens associated with different SIM cards. POR, 11. Apple asserts Chae and Narendra are incompatible because they use the same swipe for different actions. *Id.* For example, Narendra uses an upward swipe to access a password-protected screen, but Chae uses an upward swipe to access a screen associated with SIM1. *Id.* However, a compatible combination would use an upward swipe to access a password-protected screen also associated with SIM1. Indeed, Chae Figure 5 expressly discloses that exact combination, depicting an upward swipe for accessing a screen associated with SIM1 after inputting a secret code:



Chae also discloses using swipes to access non-password-protected screens associated with a different SIM2: “the second SIM (SIM2) is a SIM that does not require a user to input a secret code to unlock.” EX1005 ¶106. Cockburn admitted a POSITA could design a device that can both navigate between screens of different SIM cards and navigate between password-protected and non-password-protected screens. EX1057, 85:20-86:12. EX1056 ¶20.

A POSITA would have resolved any incompatibility

Moreover, even if there were some instances where Chae and Narendra use the same swipe for truly incompatible actions, Apple erroneously discounts a POSITA's creativity by pretending it would have been impossible to resolve the incompatibility. Pet., 12; *but see* EX1057, 13:19-14:11 (GUI design requires creativity); KSR, 550 U.S. at 421 (POSITA has ordinary creativity and is not an automaton). A POSITA would have known a user interface cannot use the same swipe movement to navigate to two different destination screens from a common origin screen. Thus, in combining Chae and Narendra, a POSITA would have resolved any trivial incompatibility by associating a unique swipe with each destination screen reachable from a common origin screen. EX1057, 88:2-7. EX1056 ¶21. Further, the Petition shows it would have been obvious to implement navigation between screens using swipes on either a multi-SIM-card device like Chae or a single-SIM-card device like Narendra. Pet., 63; EX1056 ¶16.

Topics

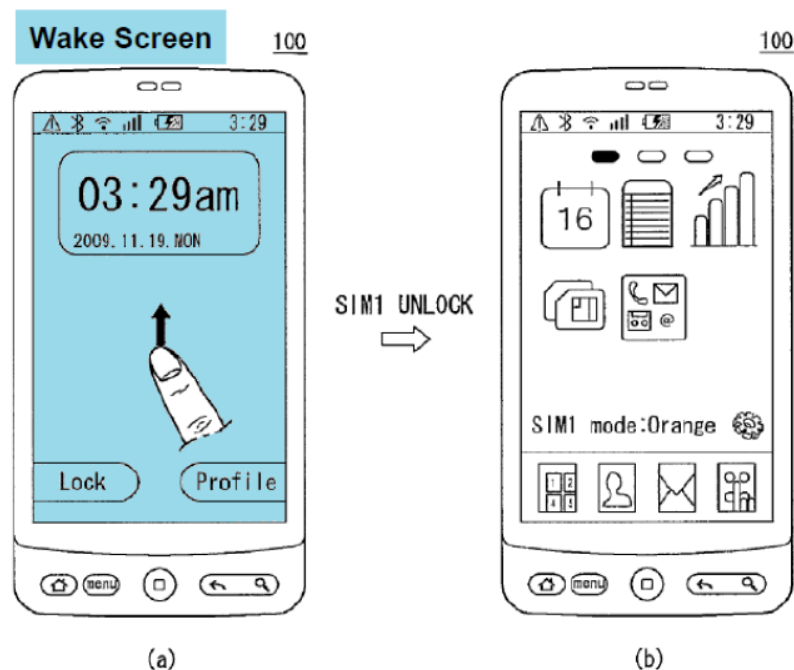
Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Claim 6 – return to wake screen with different swipe gesture

It would have been obvious to allow the user to return to the wake screen to lock the smartphone. As shown by Chae's Figure 4(a), the smartphone is or may be locked while on the wake screen.

FIG 4



EX1005, Figs. 4(a) (annotated) & 4(b). The enhanced security of locking the smartphone would have motivated a POSITA to allow the user to return to the wake screen. However, because a wake screen generally has much more limited functionality than a home screen or widget screen, a POSITA would have been motivated to provide, and would reasonably have expected success in providing, a different directional swipe gesture for returning to the wake screen than used to access the home screen or widget screen, to reduce the likelihood of user confusion or inadvertent access of the wake screen. This directional swipe gesture satisfies the claim language of a “fifth input that is directed to a portion of the home screen” and includes “fifth movement” in a “fourth direction” meeting “sixth criteria” of being “different from the first direction and the second direction.” EX1003 ¶109.

Claim 6 – return to wake screen with different swipe gesture

Even for this new theory, Masimo does not explain why it “would achieve the Petition-identified advantage of allowing a user to easily access commonly used screens from the home screen while reducing inadvertent wake-screen access.” Reply, 23. Regardless of the direction of the swipe gestures, a user would still be required to determine when certain motions cause different actions. APPLE-2003, ¶51. A particular swipe direction would not reduce the likelihood of user confusion or inadvertent access any more than a different swipe direction.

screen. However, because a wake screen generally has much more limited functionality than a home screen or widget screen, a POSITA would have been motivated to provide, and would reasonably have expected success in providing, a different directional swipe gesture for returning to the wake screen than used to access the home screen or widget screen, to reduce the likelihood of user confusion or inadvertent access of the wake screen. This directional swipe gesture satisfies the

to the wake screen. However, a POSITA would have wanted to provide a navigational path for returning to the wake screen that is easily distinguishable by the user from the usual paths for navigating to home screens and widget screens. A POSITA would have understood that the wake screen provides much more limited functionality, such as locking and unlocking the smartphone, than home screens or widget screens. Thus, a POSITA would have been motivated to provide a different directional swipe gesture for returning to the wake screen than used to access the home screen or widget screen, to reduce the likelihood of user confusion or inadvertent access of the wake screen. Further, as explained below with respect to

Claim 6 – finite design choices

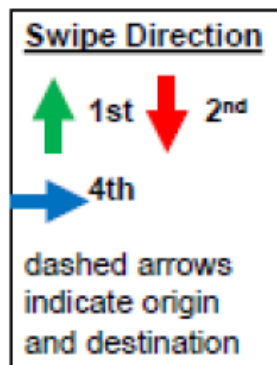
The Petition showed that selecting swipes to associate with navigational paths between screens “is a simple matter of choosing from a finite number of design choices.” Pet., 66. Apple responds, “the universe of possible swipe directions may be limited” but the number of “combinations of these directions and the various possible user interface screens is vast.” POR, 25. Apple cites nothing but a conclusory opinion of Cockburn. *Id.* (citing APPLE-2003 ¶50). No evidence supports Cockburn’s opinion that the number of combinations of possible swipe directions and screens is vast. A POSITA would have understood that useable swipe directions are practically limited to the handful users can easily distinguish and execute, such as swipes along the cardinal directions up, down, left, and right. EX1056 ¶34.

Further, a POSITA would have recognized that wake, home, widget, and control panel screens were among a small set of common user-interface screens disclosed by the prior art. Chae and Android disclose wake, home, and widget screens, and Karunamuni discloses a control panel screen, as the Petition shows. Pet., 3-5, 8, 10-11, 39-41, 45-63, 83-87. Neither the prior art nor Cockburn identify a vast number of other screens. Moreover, even if other screens existed, a POSITA would have focused on the small set of commonly used screens disclosed by the prior art to enhance ease of use for most users. Therefore, the Petition is correct that designing the navigational system of claim 6 based on the basic concepts disclosed by Chae and Narendra would have been a simple matter of choosing from a finite number of design choices. EX1056 ¶¶35-36.

Claim 6 – parallel swipe not required for motivation

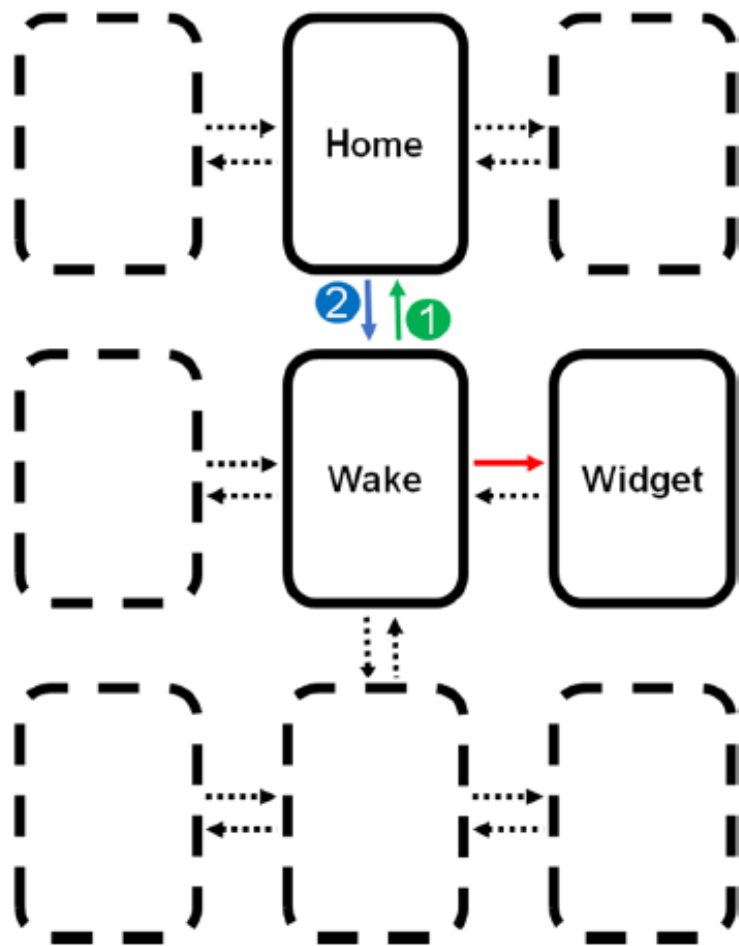
Apple also argues “it would not have been obvious to a POSITA to navigate between two screens using directions that are not parallel *and opposite* directions.” POR, 30 (emphasis in original). Apple relies on Rosenberg’s testimony that Narendra’s system of navigating between two screens using parallel and opposite swipes would have been intuitive. *Id.* (citing EX1003 ¶¶150-151). But Apple does not refute Rosenberg’s testimony that a POSITA would have been motivated to exclude the wake screen from the normal navigational path and to use a different swipe for returning to the wake screen from the home screen. EX1003 ¶109; Pet., 69-70. Specifically, because the wake screen is generally less useful than other screens, a POSITA would have wanted to reduce inadvertent wake-screen access. *Id.* EX1056 ¶44.

Claim 6 – perpendicular swipe not required



⁸ The illustrative graphics for claims 3-8 focus on illustrating available navigational paths between various user-interface screens and do not illustrate other claim limitations, which are addressed elsewhere in this Petition. Further, the graphics illustrate *general* concepts, not a *precise* navigational map required by the claims. For example, the solid arrows in the graphic above are meant to illustrate that the first, second, and fourth directions are *different* from each other, *not* that they must be a bottom-to-top swipe, a top-to-bottom swipe, and a left-to-right swipe.

Claim 6 – perpendicular swipe not required

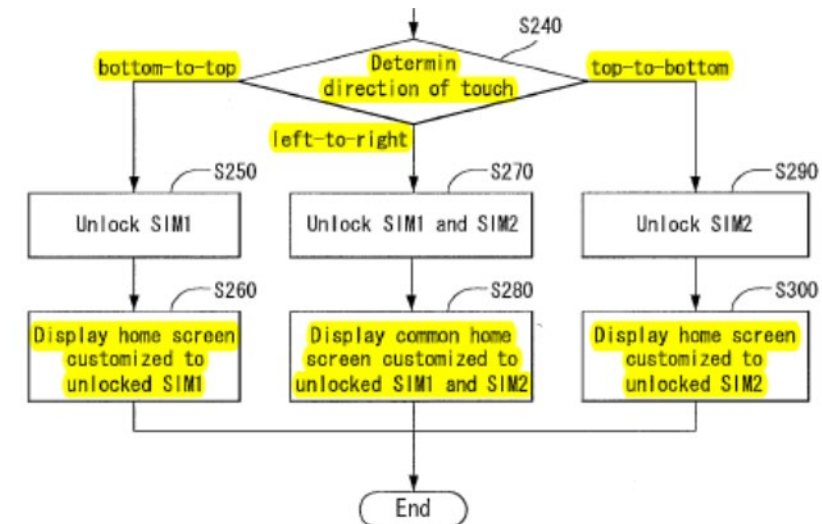


A POSITA would have understood the navigational map illustrated above would achieve the Petition-identified advantage of allowing a user to easily access commonly used screens from the home screen while reducing inadvertent wake-screen access. The user could use left and right swipes to access commonly used screens arranged in the same row as the home screen and use a downward swipe to access the less-frequently accessed wake screen. EX1056 ¶46.

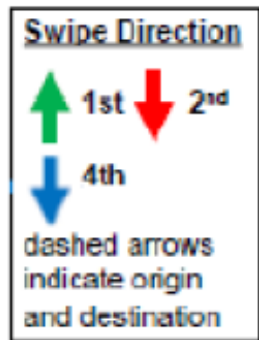
In addition, while not required by claim 6, the above-illustrated navigational map allows a user to return to the wake screen from the home screen by swiping in a direction parallel to and opposite of the direction used to navigate from the wake screen to the home screen. Therefore, Apple is incorrect to suggest that combining Chae and Narendra would necessarily require the fourth swipe for returning to the wake screen to be perpendicular to the first and second swipes for navigating from the wake screen to the home and widget screens. EX1056 ¶47.

Claim 6 – Chae does not use up all four cardinal directions

Even for swipes originating from the wake screen, Chae does not pre-define swipes for all four cardinal directions, as Apple alleges. Chae discloses at least four alternative embodiments that use swipes to unlock SIMs and display associated customized home screens including icons or widgets. None of the embodiments has pre-defined swipes in all four cardinal directions. EX1057, 133:3-10. EX1056 ¶39. Therefore, at least one cardinal direction is available for an added swipe.



Claim 6 – Apple’s proposed combination would not discourage Masimo’s combination



- Apple’s proposed combination is not Masimo’s
- Apple proposes **one** combination not embodying claim 6
- But Masimo’s combination does embody claim 6
- Proposing an alternative does not negate obviousness
- Apple does not refute that a POSITA would have desired to reduce inadvertent wake screen access

Claim 6 – a button would not discourage the combination

- Apple argues a button would be more user friendly than a swipe
- Relies on Cockburn’s subjective user-friendliness opinion
- No surveys or user-interface evaluations
- Swipes are an acceptable substitute for button or key presses and mouse movements or clicks
- POSITA would not be discouraged from using a different swipe to return to the wake screen because some users may find it easier to press a button
- “Suitable option” sufficient for obviousness. *Intel Corp. v. Qualcomm Inc.*, 21 F.4th 784, 800 (Fed. Cir. 2021); *In re Mouttet*, 686 F.3d 1322, 1334 (Fed. Cir. 2012); *In re Fulton*, 391 F.3d 1195, 1200 (Fed. Cir. 2004).

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Claim 3 – consistent swipe gesture for home screen

Shuttleworth discloses using a consistent directional swipe gesture to access the home screen from any other screen, thereby replacing hardware or virtual home buttons:

Because the Home screen can be reached so readily by a swipe from the left edge of the phone, there is no need for a dedicated hardware Home button, unlike iOS, or soft button as in Android.

EX1008 ¶353. Shuttleworth further explains:

In one implementation, the swipe is a long swipe in from the left edge; a commit gesture (e.g. a release) after the home screen appears causes the home screen to be displayed after the commit.

Id. ¶359.⁹ And Shuttleworth does not limit the swipe as necessarily coming from the left edge, also disclosing that “a swipe in from *an* edge brings up a home screen.”

Id. ¶358 (emphasis added). EX1003 ¶129.

Claim 3 – consistent swipe gesture for home screen

- “[P]roviding the user a consistent mechanism to access the home screen from any other screen was a well-known and accepted user-interface design practice for enhancing efficiency and ease of use.”
- “Using a consistent directional swipe gesture to access the home screen, whether from the wake screen or a widget screen, would have been an obvious substitute for a physical or virtual home button.”
- “[U]sing a consistent directional swipe gesture to access the home screen enhances ease of use, efficiency, and convenience by allowing the user to learn a single gesture for accessing the home screen.”

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Claim 4 – return to widget screen with different swipe gesture

As explained above, it was common to provide a dedicated directional swipe gesture, as a substitute for a physical or virtual home button, for navigation to the *home* screen. However, for *non-home* screens, it was common to provide a context-sensitive navigational system, in which the destination screen to which the user navigates in response to a particular directional swipe gesture depends on the origin screen. Narendra illustrates such a navigational system in Figure 9:

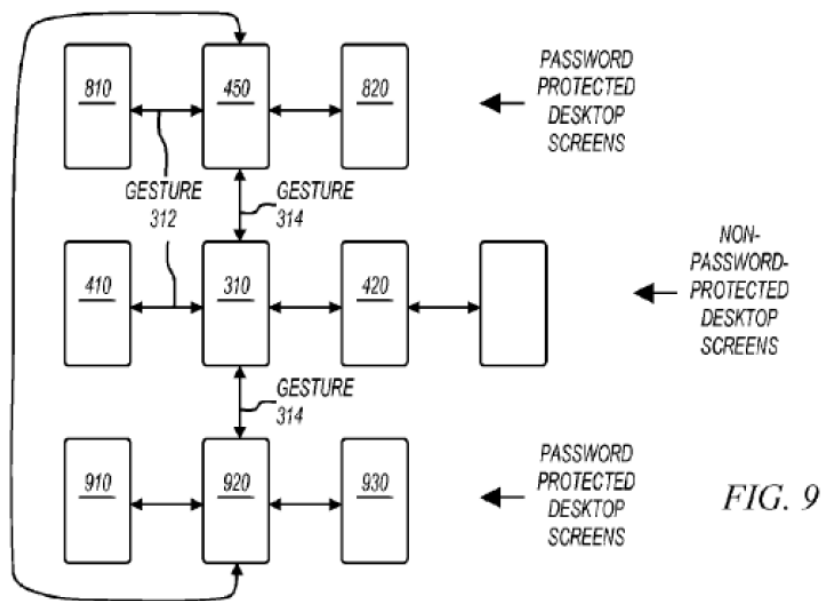


FIG. 9

Claim 4 – return to widget screen with different swipe gesture

- “[S]uch a navigational system is **easy to learn and use** because, as explained above, it is **consistent with a familiar mental model** of the current screen being part of a larger ‘map’ with additional screens located to the left, right, up, or down.”
- “[T]his navigational system would **scale easily to provide quick, efficient, and easy access** to as many screens as desired because it does not require assigning a dedicated directional swipe gesture to every screen.”
- “While recognizing the **efficiency of** providing **direct access to** a commonly accessed **home screen** using a dedicated directional swipe gesture (as a replacement for a home button), a POSITA would have understood that providing such **direct access to multiple screens would be logistically difficult and would not scale well** because each directly accessible screen would require its own dedicated directional swipe gesture.”
- “[A] POSITA would have been motivated to use a dedicated swipe for the home screen but not non-home screens to achieve: (1) the **efficiency of direct *home-screen* access** and (2) Narendra’s **scalability for *non-home* screens**.”

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Claim 5 – return to wake screen with different swipe gesture

Claim 5 is essentially the same as claim 4 except the user navigates from the home screen to the wake screen instead of to the widget screen. Simply navigating to a different screen is not a patentable distinction. Thus, as discussed for claims 3 and 4, claim 5 would have been obvious. *See supra* V.B.1-V.B.2. The directional swipe gesture in this configuration satisfies the claim language of a “fourth input that is directed to a portion of the home screen” and includes “fourth movement” in a “third direction” meeting “fifth criteria” of being “different from the first direction and the second direction.” EX1003 ¶¶141-142.

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Karunamuni discloses a control panel user interface

'352 Patent:

- Includes one or more device controls



Karunamuni:

- Discloses a smartphone control center
- Controls to change screen brightness and volume

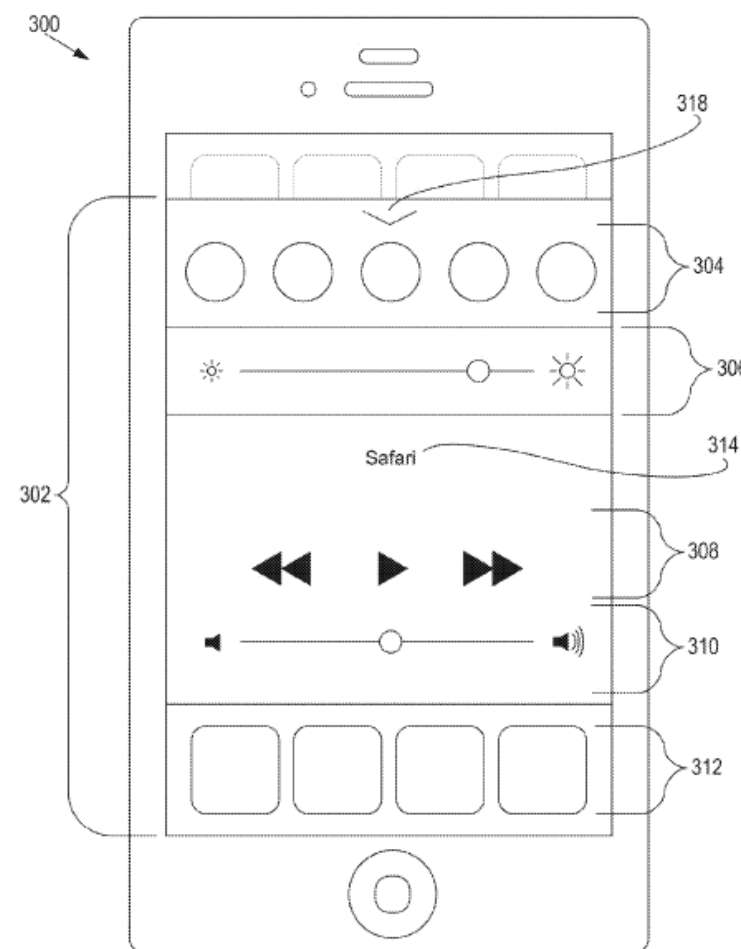


FIG. 3

Karunamuni discloses swiping to access the control panel

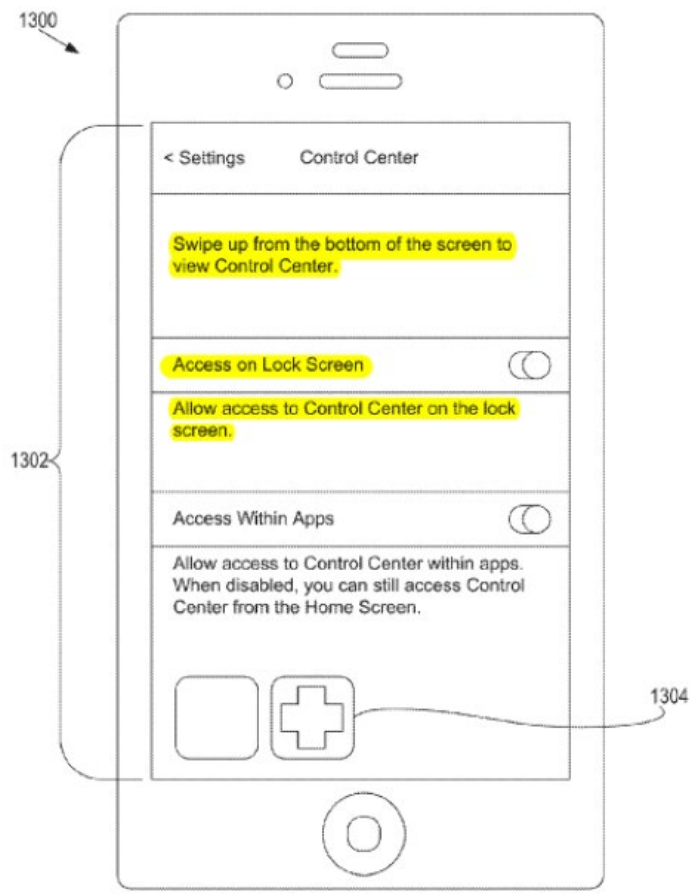


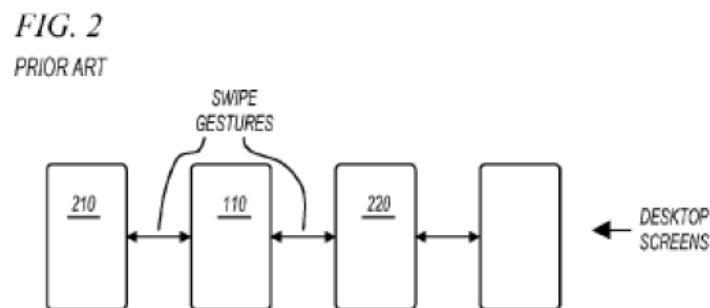
FIG. 13

Claim 7 – access control panel with different swipe gesture

- “[A] properly functioning GUI would **unambiguously** associate each gesture-invokable GUI action with a **unique gesture**.”
- “[A]ssociating gestures that are too similar to each other with different GUI actions should be avoided to **reduce the risks of user confusion and unintentional invocation** of GUI actions.”
- “Accordingly, to enhance **ease of use** and **reduce** the possibility of **unintentional invocation** of GUI actions, a POSITA would have been motivated to allow the user to **navigate to the control center** from the wake screen **by swiping in a different direction** from those used to navigate to the home screen or the widget screen from the wake screen.”

Claim 8 – return to wake screen with opposite swipe gesture

- Narendra discloses intuitive concept of swiping in opposite directions to navigate between screens.



- “Accordingly, to enhance ease of use and provide an intuitive user experience, a POSITA would have been motivated to allow the user to return to the wake screen from the control panel by swiping in the opposite direction from that used to navigate from the wake screen to the control panel.”

Topics

Obvious to:

1. Swipe in different directions for home screen and widget screen (Claim 1)
See also claims 9 and 17
2. Vary navigational paths (Dep. Claims)
 - a. Return to wake screen with different swipe gesture (Claim 6)
See also claims 14 and 22
 - b. Use consistent swipe gesture for home screen (Claim 3)
See also claims 11 and 19
 - c. Claim 3 + return to widget screen with different swipe gesture (Claim 4)
See also claims 12 and 20
 - d. Claim 3 + return to wake screen with different swipe gesture (Claim 5)
See also claims 13 and 21
 - e. Navigate to control panel (Claims 7, 8)
See also claims 15, 16, 22, and 23
3. Add limitations of substitute claims (Sub. Claims)

Tu discloses “tilt-to-wake” (claims 25-27, 30-32, and 35-37)

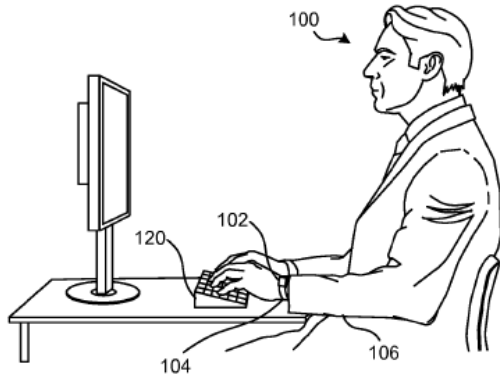


FIG. 1C

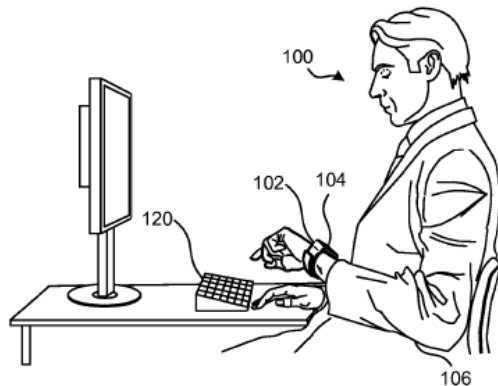


FIG. 1D

[0067] At block 410, process 400 can determine whether a possible raise gesture is identified, e.g., whether the criteria selected at block 408 are satisfied. In some embodiments, the determination can include comparing particular accelerometer data samples and/or other parameters computed from the accelerometer data samples to the threshold criteria selected at block 408, with a possible raise gesture being detected if the threshold criteria are satisfied. Further, it may be expected that if a raise gesture is being performed, upon completion of the gesture, the device will come to rest in an orientation consistent with its display being in the user's line of sight. Accordingly, block 410 can include estimating an ending pose for the device, e.g., based on the newest samples read from the buffer, which can correspond to the end of the interval. The ending pose can be compared to a range of poses consistent with the display being in the user's line of sight (e.g., a range of tilt angles in the y and/or x directions relative to horizontal), with a possible raise gesture being detected if the ending pose is within this range. The term “focus pose” is used herein to refer generally to any pose within the range poses deemed consistent with the device being in the user's line of sight (e.g., based on tilt angles).

[0069] If, at block 410, a possible raise gesture is detected, then at block 412, process 400 can notify wake control logic 230 that a possible raise gesture was detected. As described below, wake control logic 230 can take action in response to this notification, such as notifying applications processor 204 and/or user interface 202. At block 414, process 400 can monitor additional accelerometer data samples (e.g., including newly received samples outside the original window) to detect whether the device dwells (e.g., holds an approximately constant position) in a focus pose or continues to move. For example, as noted above, focus pose can be defined based on a particular range of tilt angles expected while a user is looking at the device. Tilt angles can be inferred from accelerometer data, and if the tilt angle reaches and remains within this range, that condition can correspond to dwell in the focus pose.

[0070] In some embodiments, the range of accelerometer data values corresponding to focus pose can depend at least in part on the starting pose. For instance, the transition from a starting pose to a focus pose can involve accruing a certain amount of tilt along the y axis (also referred to herein as “y-tilt”). For instance, accrued y-tilt can be measured as a net change from a starting a_y component that is greater than or equal to zero (while the user's wrist is turned away) to an ending a_y component that is less than zero (when the wrist is turned to orient the display toward the user's eyes). Block 414 can include monitoring whether the y-tilt remains constant, continues to accrue, or reverses direction, and a loss of accrued y-tilt can indicate moving out of focus pose.

Motivation to implement tilt-to-wake

- “A POSITA would have been motivated to implement the ‘tilt-to-wake’ functionality disclosed by Tu with the devices disclosed by the prior art cited in the Petition to manage the device’s power consumption more efficiently and improve ease-of-use.”
- “A POSITA would have understood that tilting the device would be a more convenient alternative to a button press.”
- Tu suggests implementing its raise gestures “in other portable devices, such as mobile phones, tablets, or the like ...”
- Initial Apple Watch had both “raise to wake” and button press to wake.
- Tu teaches how to reduce false positives by tuning “sensitivity.”
- No teaching away, in Chae or otherwise, from tilt-to-wake

Apple's Dascola Publication

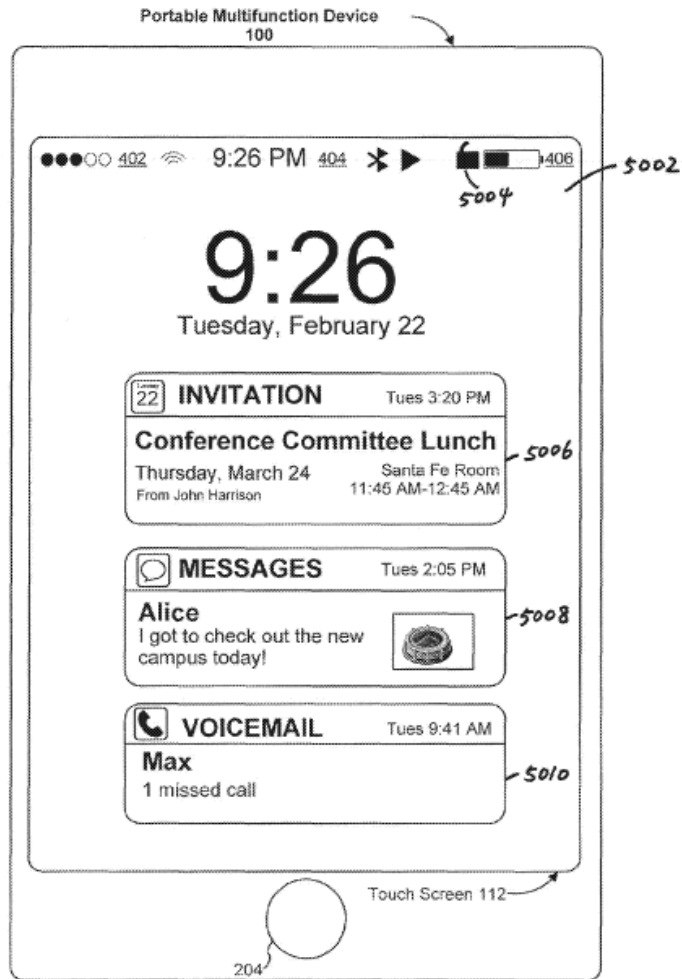


Figure 5A1

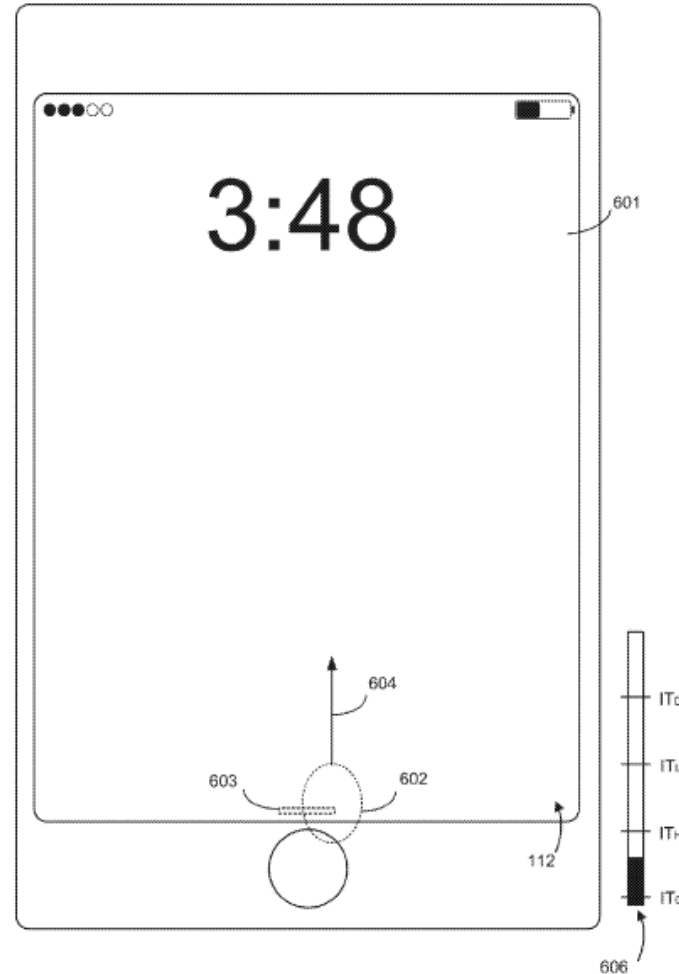


Figure 6A

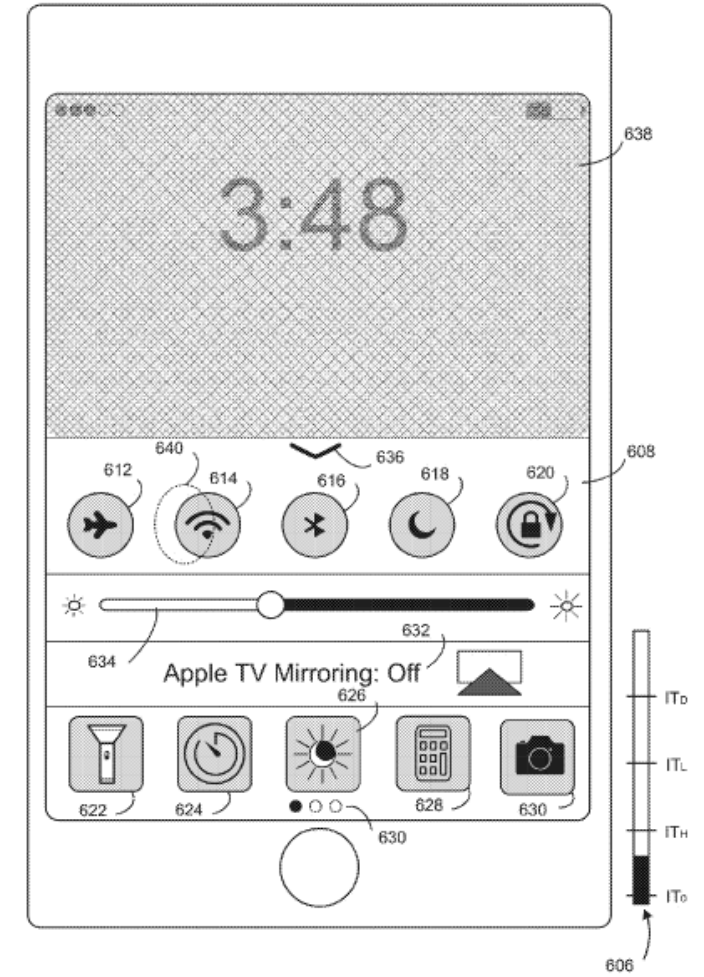


Figure 6B

Slide wake screen to reveal underlying UI (claims 28, 33, 38)

Dascola: “control user interface 608 is gradually revealed ...”

Brinda:

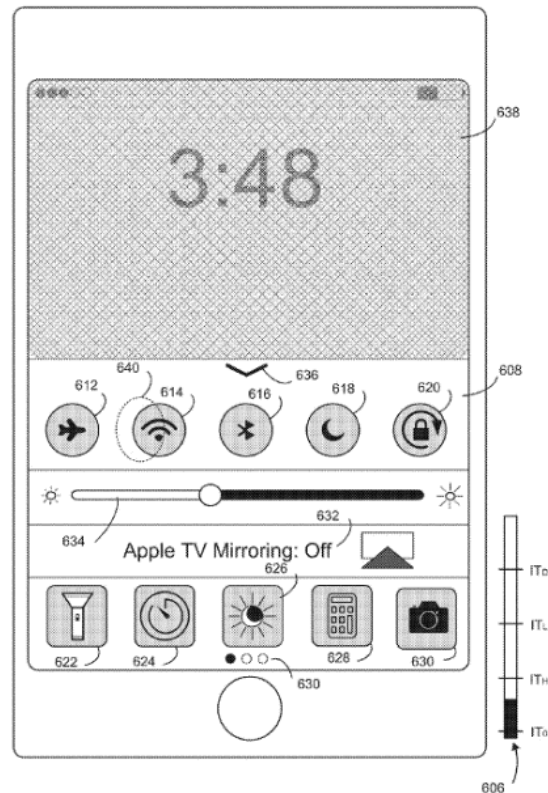


Figure 6B

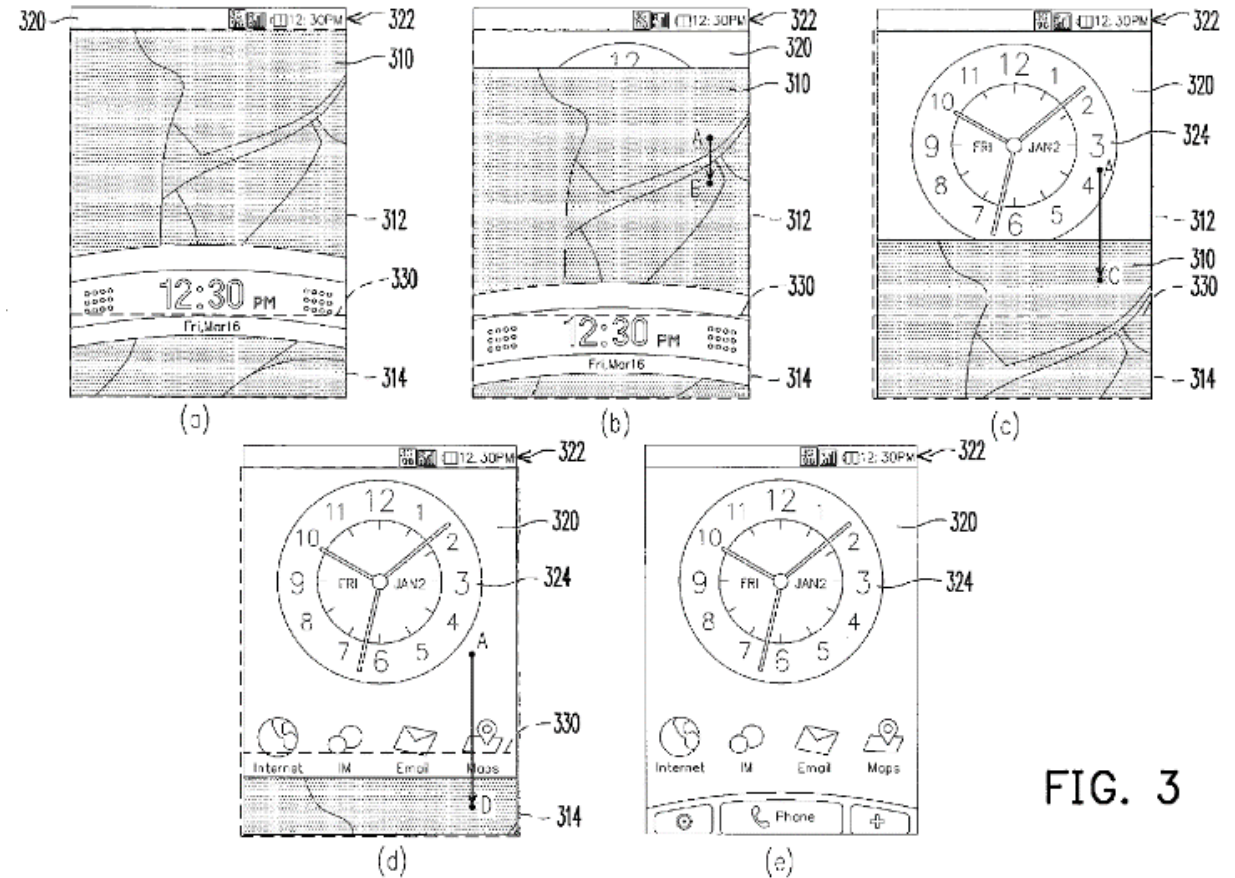


FIG. 3

Motivation to slide wake screen up to reveal underlying UI

- “A POSITA would have been motivated to implement the sliding wake screen disclosed by Dascola and Brinda with the devices disclosed by the prior art in the Petition to **improve the aesthetics** of a user interface.”
- **Chae discloses bottom-to-top input** on wake screen to display another screen.
- “[A]nimating the transition between screens, such as by sliding the wake screen up to gradually reveal another screen would have been a **trivial change**.”
- Layering windows to create **3D** user interfaces was **familiar** and **popular**.
- Improved aesthetics motivation appropriately focuses on the **claimed invention**. *In re Urbanski*, 809 F.3d 1237, 1243-1244 (Fed. Cir. 2016); *Janssen Pharm., Inc. v. Teva Pharm. Inc.*, 97 F.4th 915, 925 (Fed. Cir. 2024); *Intelligent Bio-Sys., Inc. v. Illumina Cambridge Ltd.*, 821 F.3d 1359, 1367 (Fed. Cir. 2016).
- PTAB has found improved **aesthetics an appropriate motivation**. *Kawasaki Rail Car, Inc. v. Blair*, IPR2017-00117, Paper 44 at 27-28 (PTAB May 2, 2018).

Power button dismisses wake screen and notifications

(claims 29, 34, 39)

- Dascola discloses wake-screen notifications.
- Power buttons were a ubiquitous feature of mobile devices.
- Dascola discloses a “push button 206 for powering the device on/off and locking the device.”
- Lee discloses “[t]he power button 6 may be used to toggle the display screen 2 on and off (i.e. transition the device 1 between wake and sleep states).”
- Lee discloses “the lock screen and associated widget is not viewable” when the display is off, suggesting notifications are not viewable when the display is off.
- Notifications on wake screen provide “easier access” and ensure that “users see the notifications, thereby improving the user experience.”
- Implementing a power button provides a “common element ... that users would have expected a device to have.”