

Apple Inc. (Petitioner)
v.
Smith Interface Technologies, LLC (Patent
Owner)
Petitioner Demonstratives

Case No. IPR2024-01084
U.S. Patent No. 10,649,578

Before Hon. Jameson Lee, Jeffrey S. Smith, Sharon Fenick
Administrative Patent Judges

Table of Contents

Overview of the Instituted Grounds	3
Background Info: The '578 Patent and Prior Art	5
Issue 1: Shiplacoff Renders Obvious Transitioning from Full Screen Mode to the Launcher with a Tap or Flick	9
Issue 2: Shiplacoff in view of Nan Underscores Obviousness of Transitioning with Tap or Flick	30
Issue 3: The Dependent Claims are Similarly Obvious	38

Overview of the Instituted Grounds

Overview of the Instituted Grounds

Ground	Claims	Basis for Rejection (35 U.S.C. § 103)
1	1-4, 8-18, 22-30, and 34-39	Shiplacoff
2	1-4, 8-18, 22-30, and 34-39	Shiplacoff in view of Nan

Pet., 2; see Inst. Dec., 4

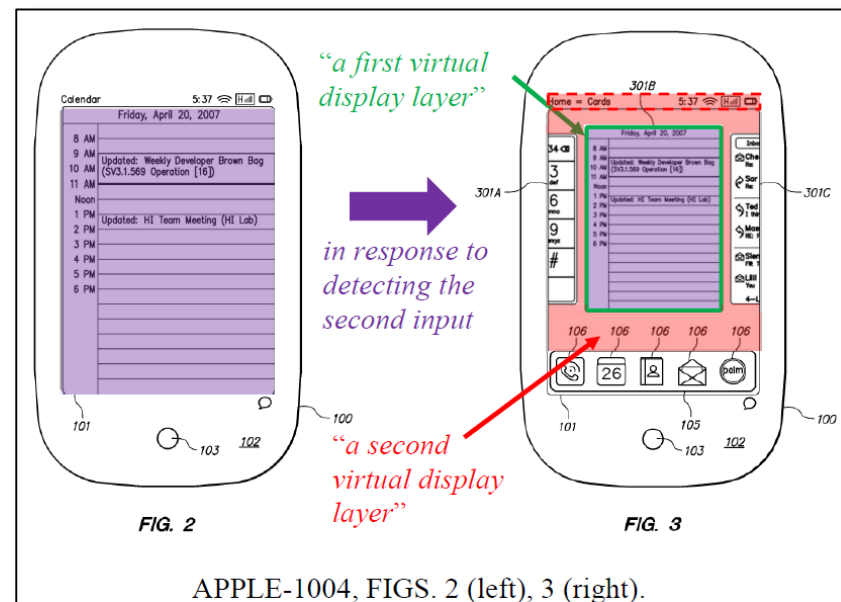
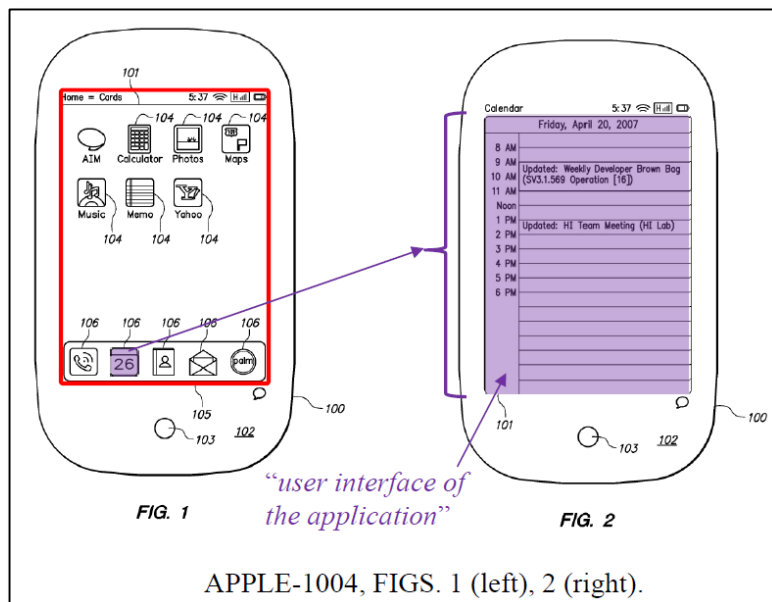
Background Info: The '578 Patent and Prior Art

Exemplary Claim 1 of the '578 Patent

1. An electronic device, comprising:
a display;
a touch interface;
one or more processors;
memory; and
one or more programs, wherein the one or more programs are stored in the memory and configured to be executed by the one or more processors, the one or more programs including instructions for:
displaying a home screen on the display, the home screen including an icon associated with an application;
while displaying the home screen, detecting a first input by a first contact on the icon;
in response to detecting the first input, replacing the home screen with a user interface of the application;
while displaying the user interface of the application, detecting a second input by a second contact that includes movement across the display in a direction;

in response to detecting the second input and in accordance with a determination that the second input meets one or more criteria that is met when the second input is detected to include a movement parameter that is above a movement threshold, displaying at least a portion of the user interface of the application in a first virtual display layer that appears at a lesser depth as compared to a second virtual display layer, such that the at least portion of the user interface of the application is reduced in size and is further displayed in its entirety when displayed in the first virtual display layer; and
in response to detecting the second input and in accordance with a determination that the second input does not meet the one or more criteria that is met when the second input is detected to include the movement parameter that is above the movement threshold, replacing the user interface of the application with the home screen including the icon associated with the application.

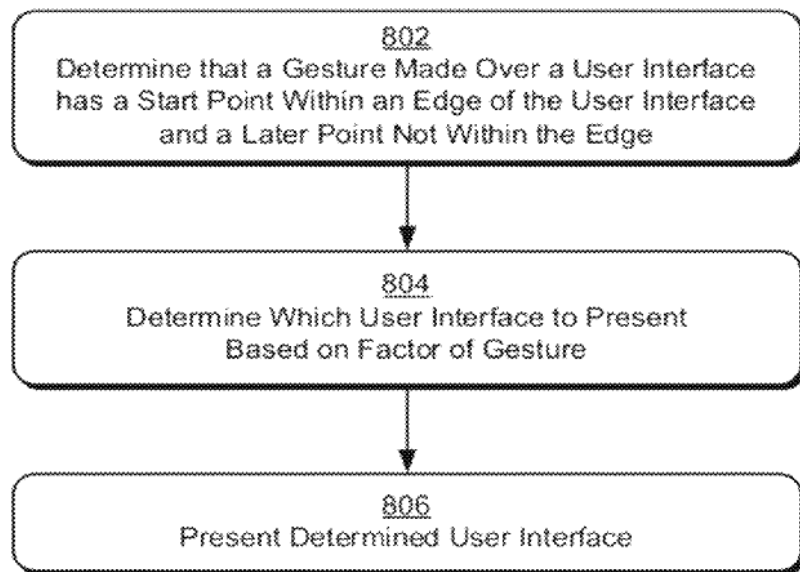
Shiplacoff (APPLE-1004)



APPLE-1004, FIGS. 2-3

Pet., 15, 18

Nan (APPLE-1005)



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., single-finger or multi-finger), or a speed of the gesture. In some cases, two or more factors of a gesture determine which interface to present such as a drag-and-hold gesture having a drag length and hold time or a drag-and-drop gesture having a drag length and drop position. 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, speed, or a number of inputs (e.g., fingers).

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 **124** 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**.

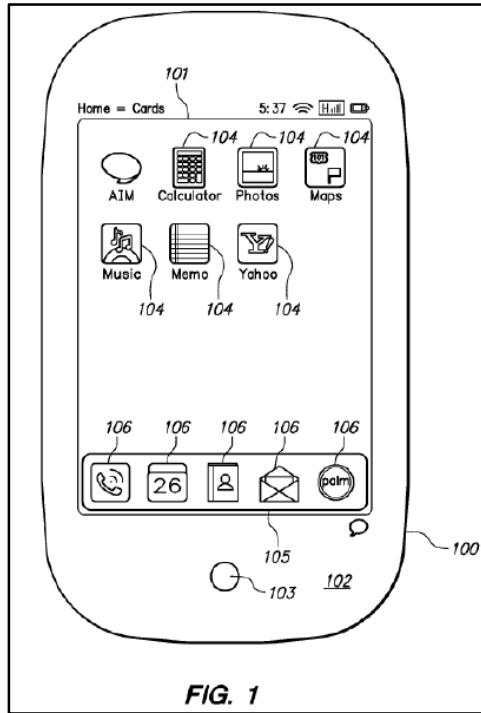
APPLE-1005, FIG. 8, 7:30-46, 9:9-21

Pet., 42-43, 46-47

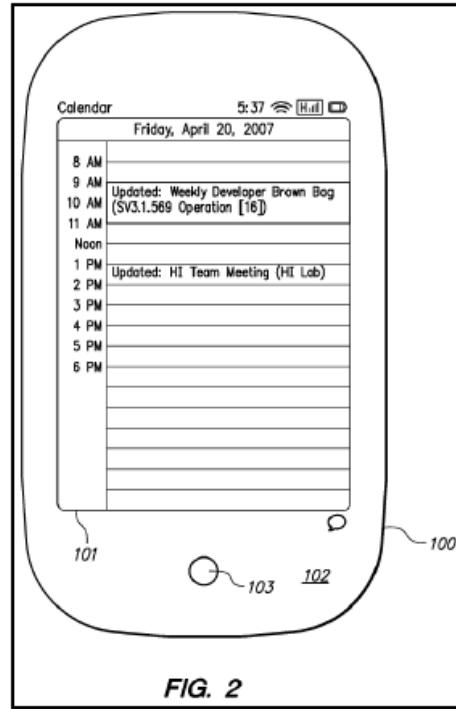
Issue 1

Shiplacoff Renders Obvious Transitioning
from Full Screen Mode to
the Launcher with a Tap or Flick

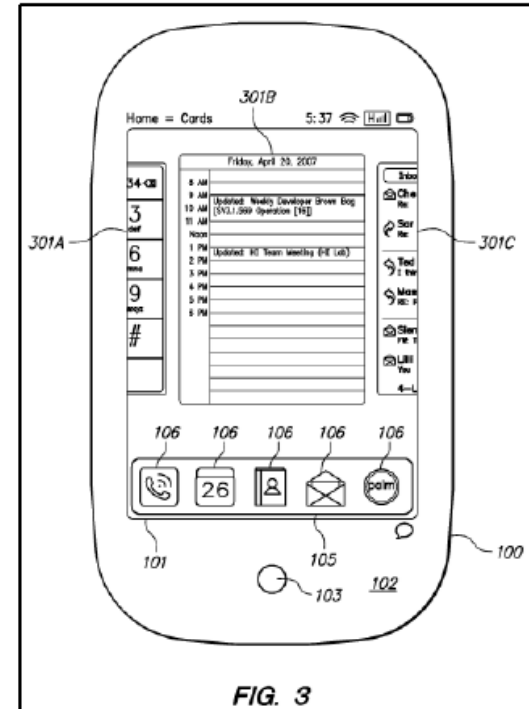
Shiplacoff Teaches Multiple Display Modes



Launcher Mode



Full screen mode



Card Mode

APPLE-1004, FIGS. 1-3
Pet., 7, 9

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Shiplacoff Teaches Various Mechanisms, Including Gestures, for Switching Between Modes

Switching Among Modes

[0124] In one embodiment, the user can switch among modes pressing physical button 103 or some other button on device 100. In another embodiment, the user can touch or double-tap a card 301 in shuffle mode to initiate a switch to full-screen mode or to card mode. In yet another embodiment, device 100 switches from shuffle mode to full-screen mode (or to card mode) after some period of inactivity. Other mechanisms can also be provided for switching among the modes. In one embodiment, a transition effect is performed when switching from one mode to another.

[0125] In another embodiment, the user can switch among card mode, shuffle mode, and grid mode by tapping in a designated area on screen 101. In one embodiment, the designated area is any area not occupied by any card 301. In another embodiment, the designated area is an area below cards 301.

[0126] In yet another embodiment, when in card mode, tapping the area below the row of cards causes a switch to shuffle mode, whereas, when in shuffle mode, tapping any area not occupied by a card 301 causes a switch to card mode. In other embodiments, other user input actions can cause a switch between the modes, including for example, entering a gesture, typing a key on a keyboard, pressing a button, selecting a menu command, activating an on-screen button or other element, or the like.

APPLE-1004, [0124]-[0126];

Pet. 15-16, 21-24; Reply 1-2, 4-5

Shiplacoff Teaches Transitioning Between Modes With Various Gestures

[0127] In another embodiment, the user can switch among the various modes by performing a gesture. One example of such a gesture is a “swipe up” gesture beginning in gesture area 102 and extending onto screen 101. In one embodiment, such a gesture switches from full-screen mode to card mode, or from card mode to shuffle mode, or from shuffle mode to grid mode. In another embodiment, such a gesture switches from full-screen mode to card mode, or from card mode to grid mode. In yet another embodiment, such a gesture switches from full-screen mode to card mode, or from card mode to a launcher screen.

[0128] Another example of a mode change gesture is a “swipe down” gesture beginning on screen 101 and extending onto gesture area 102. In one embodiment, such a gesture switches from grid mode to shuffle mode, or from shuffle mode to card mode, or from card mode to full-screen mode. In another embodiment, such a gesture switches from grid mode to card mode, or from card mode to full-screen mode. In yet another embodiment, such a gesture switches from a launcher screen to card mode, or from card mode to full-screen mode.

[0129] In other embodiments, other gestures can be used for mode changes; for example, a multi-finger gesture such as a pinch can be used to change from one mode to another.

APPLE-1004, [0127]-[0129];

Pet. 15-16, 21-24; Reply 1-2, 4-5

Shiplacoff Teaches Many Different Gestures

Unquestioned Testimony of Apple's Expert (Dr. Terveen)

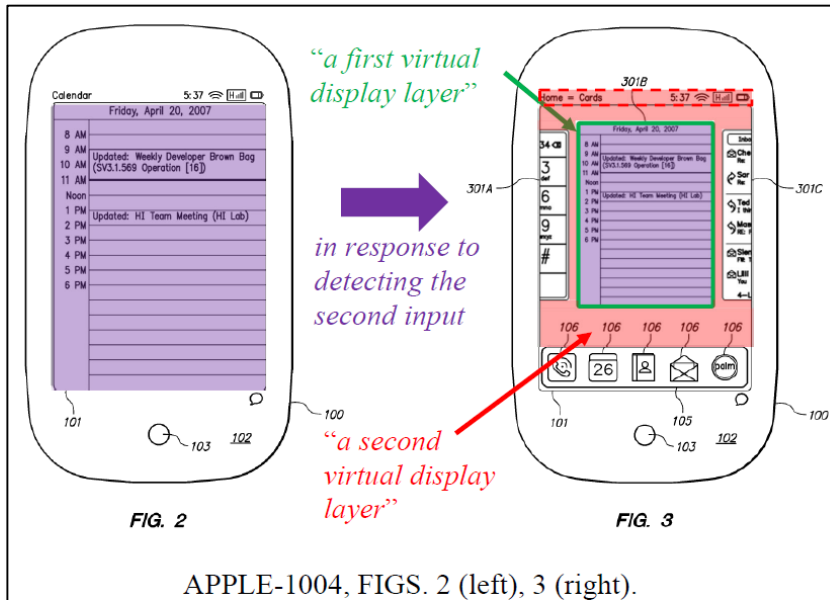
² Indeed, Shiplacoff teaches its system is capable of recognizing and distinguishing numerous gestures, including: “tapping;” “double-tap;” “dragging;” “tapping and holding;” “flicking;” “swipe up;” “swipe down;” and “multi-finger gesture such as a pinch.” *See, e.g.*, APPLE-1004, [0077], [0084], [0091], [0098], [0100], [0104], [0112], [0113], [0124]-[0129], [0169]. Almost all of these gestures are described as causing multiple possible outcomes depending at least on where they are performed on the screen and the context of what’s being displayed at the time the gesture is received. *See id.*

Shiplacoff

[0132] One skilled in the art will recognize that mode changes can take place in response to other types of commands, contexts, and/or environmental factors.

APPLE-1032, p. 10, n. 2; APPLE-1004, [0132];
Pet. 21-24; Reply 1-2, 4, 7-11

Example of Transitioning with “Swipe-Up”



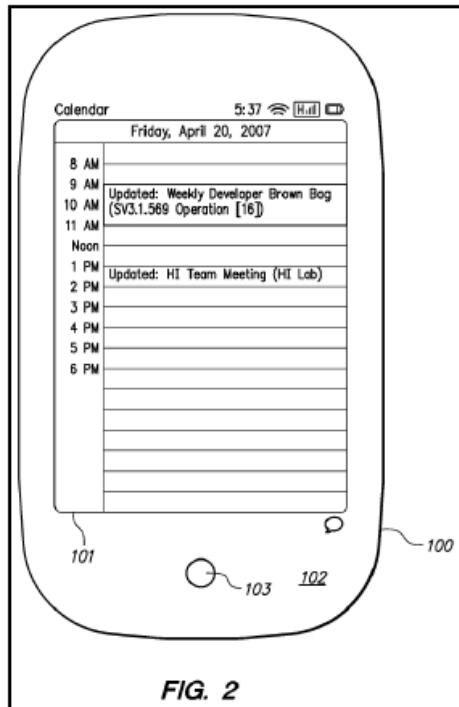
[0127] In another embodiment, the user can switch among the various modes by performing a gesture. One example of such a gesture is a “swipe up” gesture beginning in gesture area 102 and extending onto screen 101. In one embodiment, such a gesture switches from full-screen mode to card mode, or from card mode to shuffle mode, or from shuffle mode to grid mode. In another embodiment, such a gesture switches from full-screen mode to card mode, or from card mode to grid mode. In yet another embodiment, such a gesture switches from full-screen mode to card mode, or from card mode to a launcher screen.

Shiplacoff Teaches Motivation for Returning to Home Screen

[0099] In one embodiment, the user can launch additional activities by returning to the launch screen as shown in FIG. 1. In one embodiment, this is performed by performing a gesture or other command. In one embodiment, the user can also launch additional activities by touching an icon **106** in quick-launch area **105**. In one embodiment, newly launched activities generate new cards that are added to the end of the sequence of cards shown in FIG. 7. In one embodiment, the card for the newly launched activity is, by default, given focus and presented in the central position on screen **101**.

APPLE-1004, [0099];
Pet., 23-24; Reply, 9

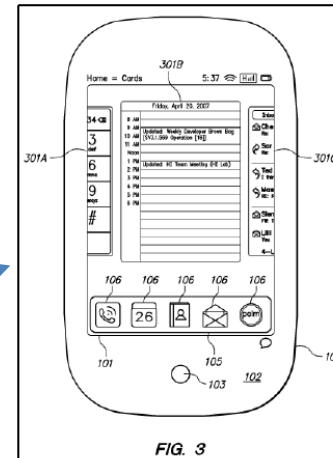
Petition's Explanation of Switching Modes



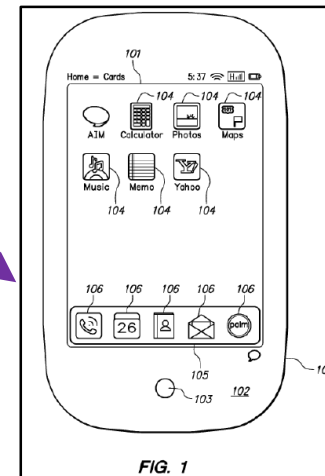
Full screen mode

Swipe-up

"different gesture
such as tap or flick"



Card Mode



Launcher Mode

APPLE-1004, FIGS. 1-3
Pet., 17-24; Reply 6-7

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

PO Acknowledges Different Functions Need Different Gestures

Shiplacoff

[0129] In other embodiments, other gestures can be used for mode changes; for example, a multi-finger gesture such as a pinch can be used to change from one mode to another.

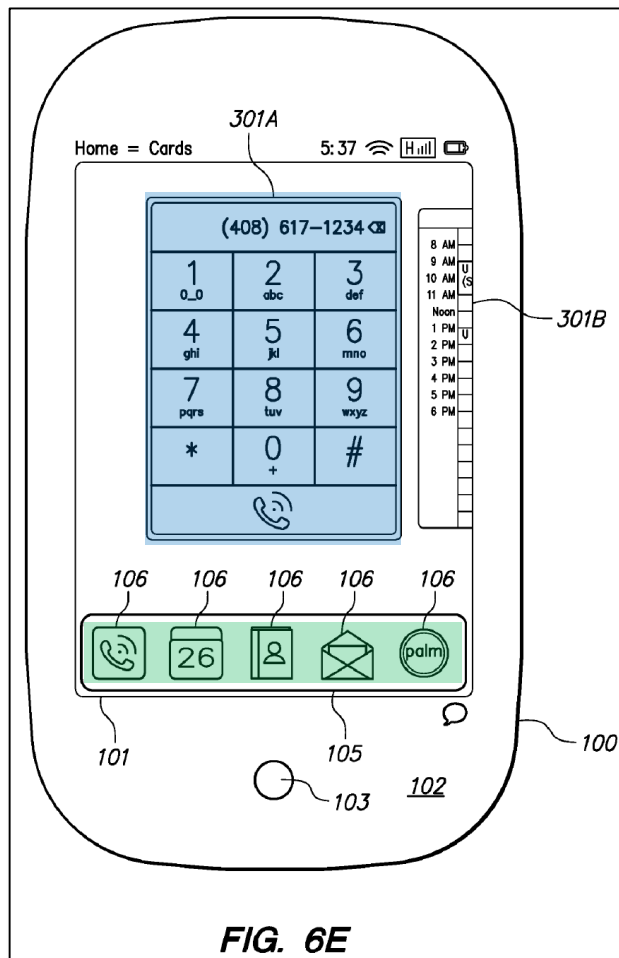
APPLE-1004, [0129]
Reply, 10-11

POR

A POSITA would have recognized that, to program a device such that a particular input triggers a switch from “full-screen mode” to either the “launch screen” or one of the other display modes, the input must not already be programmed to trigger a different action for manipulation of on-screen elements in the application. EX2004, ¶93; *see also* EX2006, 42:16-45:17 (agreeing that, “[i]n any given user interface or mode, a gesture should have a single meaning.”)

POR, 49

Shiplacoff Teaches Examples of Same Gesture Type Causing Different Outcomes Based on Different Characteristics



[0098] Once the telephone application is in the central position in card 301A as shown in FIG. 6E, the user can tap on card 301A or press button 103 to switch to full-screen mode, as shown in FIG. 6F. In one embodiment, the user can also interact with the telephone application directly in card mode as shown in FIG. 6E.

[0099] In one embodiment, the user can launch additional activities by returning to the launch screen as shown in FIG. 1. In one embodiment, this is performed by performing a gesture or other command. In one embodiment, the user can also launch additional activities by touching an icon 106 in quick-launch area 105. In one embodiment, newly launched activities generate new cards that are added to the end of the sequence of cards shown in FIG. 7. In one embodiment, the card for the newly launched activity is, by default, given focus and presented in the central position on screen 101.

[0126] In yet another embodiment, when in card mode, tapping the area below the row of cards causes a switch to shuffle mode, whereas, when in shuffle mode, tapping any area not occupied by a card 301 causes a switch to card mode.

APPLE-1004, FIG. 6E, [0098]-[0099]
Reply, 11-12

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

To Transition to Launcher, Would Have Used Gesture With Different Parameters Than “Swipe Up”

Testimony of Dr. Terveen

61. When a gesture different from the swipe up gesture is used, it would also have been obvious to a POSITA that such a gesture would not have the same characteristics and movement parameters as the swipe up gesture so that Shiplacoff's device can distinguish different gestures. For example, it would have been obvious, if not understood, to use a different gesture, such as a tap or flick, that does not have the same movement parameters as those of a swipe up gesture (e.g., touch and slide action for a certain displacement amount and upward direction).

APPLE-1003, ¶61;
Pet., 23-24

Duarte-992 Confirms Small Changes in Gesture Characteristics Result in Different Associated Actions

[0069] *Scroll:* Referring now to Fig. 4H, the user can perform a drag-scroll operation by performing a drag gesture 402H (also referred to as a flick gesture) across screen 101 in a direction that supports scrolling for the current state of the display. In one embodiment, the drag must start moving immediately upon contact with screen 101, in order to be recognized as a scroll. The current display scrolls by an amount proportional to the distance moved by the user's finger, as shown in the right side of Fig. 4H. In one embodiment, the scroll amount can be adjusted or dampened as appropriate.

[0070] The user can also flick across screen 101 in a direction that supports scrolling for the current state of the display. In one embodiment, the flick must start immediately upon contact with screen 101, and the user's finger must leave the surface of screen 101 before stopping movement, in order to be recognized as a flick. The current display scrolls by an amount proportional to the speed and distance which the user flicked. A drag-scroll may be converted into a flick by lifting the finger before coming to a rest.

SMITH-2003, [0069]-[0070]
Reply, 13-14

Shiplacoff Differentiates Gestures Based on “When, Where, and How” Gesture Input

Testimony of Dr. Terveen

19. Based on these teachings, a POSITA would have understood or at least found obvious that Shiplacoff’s device would have been able to distinguish between distinct instances of “flick” gestures depending on when, where, and how a given gesture was input by the user. For example, like the “swipe up” gesture described for changing modes, a “flick” gesture started in the gesture area 102 or near an edge of screen 101 would have been distinguishable from a “flick” gesture started well within screen 101 and its display of a particular activity. *See* APPLE-1004, [0127]; *see also id.*, [0064] (describing use of the “touch-sensitive gesture area 102 for entering gesture-based commands”). In fact, Dr. Madisetti acknowledges that Shiplacoff is able to at least distinguish gestures that begin in gesture area 102 and cross over onto screen 101 from those that start on screen 101. SMITH-2004, [78]-[79], [99] (acknowledging that Shiplacoff “discloses that

APPLE-1032, ¶19

Reply, 14-15

Shiplacoff and Duarte-992 Teach Additional Mechanisms to Differentiate Gestures

Shiplacoff

[0064] In one embodiment, screen 101 is touch-sensitive, so that a user can activate an application or function by touching a displayed icon 104, 106. In one embodiment, device 100 also includes a touch-sensitive gesture area 102 for entering gesture-based commands, as described in the above-referenced related application. In another embodiment, screen 101 and/or gesture area 102 is sensitive to user motion in proximity to the surface, so as to detect gestures and commands performed near the surface but without necessarily contacting the surface.

Duarte-992

[0064] In one embodiment, a modifier key, such as a shift key, is provided. Modifier key may be physical button 103, or some other button (not shown). Certain commands performed in touch-sensitive screen 101 can be modified by performing the command while holding down the modifier key, or by pressing the modifier key prior to performing the command.

APPLE-1004, [0064]; SMITH-2003, [0064]

Reply, 14-15

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Issue 1 (cont.)

Smith's Remaining Arguments
Are Equally Unavailing

A “Tap” Includes Movement, e.g., Below a Threshold

Rebuttal to:

“...distinguishing between an input that does not have movement (such as a tap) and one that does have movement (such as a swipe) is insufficient...”
POR, 39

Petition

Shiplacoff’s device can distinguish different gestures. APPLE-1003, ¶[61]. For example, it would have been obvious, if not understood, to use a different gesture, such as a tap or flick, that does not have the same movement parameters as those of a swipe up gesture (e.g., touch and slide action for a certain displacement amount and upward direction). APPLE-1003, ¶[61]. Accordingly, it would have been

Pet., 23-24

A “Tap” Includes Movement, e.g., Below a Threshold

**Rebuttal
to:**

“...distinguishing between an input that does not have movement (such as a tap) and one that does have movement (such as a swipe) is insufficient...”
POR, 39

Shaffer (APPLE-1008)

For example, a tap gesture may be effectively defined by a sequence of two, or optionally, three sub-events. First, a touch should be detected, and this will be sub-event 1. For example, the touch sub-event may be a user’s finger touching a touch-sensitive surface in a view that includes the event recognizer having state machine 400. Second, an optional measured delay where the touch does not substantially move in any given direction (e.g., any movement of the touch position is less than a predefined threshold, which may be measured as a distance (e.g., 5 mm) or as a number of pixels (e.g., 5 pixels) on the display), and the delay is sufficiently short, would serve as sub-event 2. Finally, termination of the touch (e.g., liftoff of the user’s finger from the touch-sensitive surface) will serve as sub-event 3. By coding the event recognizer state machine 400 to transition between states based upon receiving these sub-events, the event recognizer state machine 400 effectively expresses a tap gesture event definition. However,

APPLE-1008, 28:37-62

Reply, 16-17;

see also APPLE-1032, ¶¶23-26

(Dr. Terveen endorsing same)

A “Tap” Includes Movement, e.g., Below a Threshold

Rebuttal to:

“...distinguishing between an input that does not have movement (such as a tap) and one that does have movement (such as a swipe) is insufficient...”
POR, 39

Matthews (APPLE-1010)

The gesture engine 280 is configured to determine tells which gesture is being performed by the user at the computing device 215. For instance, the gesture engine 280 is capable of processing a continuous, uninterrupted movement of a gesture and/or a duration of the gesture utilizing a timer. If the duration of the gesture does not comprise a distance of uninterrupted movement above a predefined distance or does not meet a predefined threshold time period, the gesture is considered a tap-type operation (e.g., tap simulating a mouse click). Otherwise, the gesture is considered a drag operation.

APPLE-1010, 10:65-11:7

Reply, 17-18;

see also APPLE-1032, ¶¶23-26

(Dr. Terveen endorsing same)

Shiplacoff and Duarte Support Multiple Uses of Gesture Type Based on Differences in Characteristics

Rebuttal to:

“Shiplacoff (via Duarte-992) discloses that its ‘flick’ is an on-screen input that, when in ‘full-screen mode,’ is already programmed to trigger ‘scrolling’ of displayed objects.”
Sur-reply, 7

Original Testimony of Apple’s Expert (Dr. Terveen)

been obvious to a POSITA that Shiplacoff also renders obvious this feature. For instance, Shiplacoff’s device detects user input and differentiates between contacts and gestures such as a single tap or swipe up gesture, so that it can determine the appropriate corresponding operation to execute. A POSITA would have understood

APPLE-1003, ¶58
Pet., 21-22

Dr. Terveen Reply Testimony

19. Based on these teachings, a POSITA would have understood or at least found obvious that Shiplacoff’s device would have been able to distinguish between distinct instances of “flick” gestures depending on when, where, and how a given gesture was input by the user. For example, like the “swipe up” gesture described for changing modes, a “flick” gesture started in the gesture area 102 or near an edge of screen 101 would have been distinguishable from a “flick” gesture started well within screen 101 and its display of a particular activity. *See* APPLE-1004, [0127]; *see also id.*, [0064] (describing use of the “touch-sensitive gesture area 102 for entering gesture-based commands”). In fact, Dr. Madisetti

APPLE-1032, ¶19
Reply, 14-15

Shiplacoff and Duarte Support Multiple Uses of Gesture Type Based on Differences in Characteristics

Rebuttal to:

“Shiplacoff (via Duarte-992) discloses that its ‘flick’ is an on-screen input that, when in ‘full-screen mode,’ is already programmed to trigger ‘scrolling’ of displayed objects.”
Sur-reply, 7

Original Testimony of Apple’s Expert (Dr. Terveen)

threshold”). APPLE-1004, ¶[127]. For instance, a POSITA would have understood that some of the differences between a flick up gesture and tap gesture, on the one hand, and a swipe up gesture, on the other are the speed and movement of the gestures. APPLE-1004, ¶¶[100], [127]. Thus, while the gestures may have an initial point of user input contact and a final point of input contact, the difference in the gestures include the time duration of the touch (which then determines the speed), the length of the movement after the initial touch, and the directionality of the movement after the initial touch. In general, it was known in the art that different gestures can be configured by parameters such as a speed, location, time, and direction of touches. APPLE-1008³, 19:63-65 (“event information for a swipe event includes two or more of: a location, time stamp, direction, and speed of a swipe”), 18:41-43; APPLE-1009⁴, 18:1-40 (Tables 1 and 2 depicting a time period of 150 ms associated with a swipe gesture); Moreover, a minimum predetermined time period

APPLE-1003, ¶80
Pet., 36-37

Petition Relied Upon Gesture With Different “Movement Parameter” from Swipe-Up, Without Changing Start Point

Rebuttal to:

“Petitioner’s mapping is premised on using the exact same input on the same object (i.e., Shiplacoff’s ‘flick’) to trigger different functions.” Sur-reply, 7

61. When a gesture different from the swipe up gesture is used, it would also have been obvious to a POSITA that such a gesture would not have the same characteristics and movement parameters as the swipe up gesture so that Shiplacoff’s device can distinguish different gestures. For example, it would have been obvious, if not understood, to use a different gesture, such as a tap or flick, that does not have the same movement parameters as those of a swipe up gesture (e.g., touch and slide action for a certain displacement amount and upward direction). Accordingly, it

APPLE-1003, ¶¶61;
Pet., 23-24

Issue 2

Shiplacoff in view of Nan Underscores
Obviousness of Transitioning Based on
Gestures with Different Movement Parameters

Nan Explicitly Teaches Using Gestures with Different Movement Parameters to Switch Between Interfaces

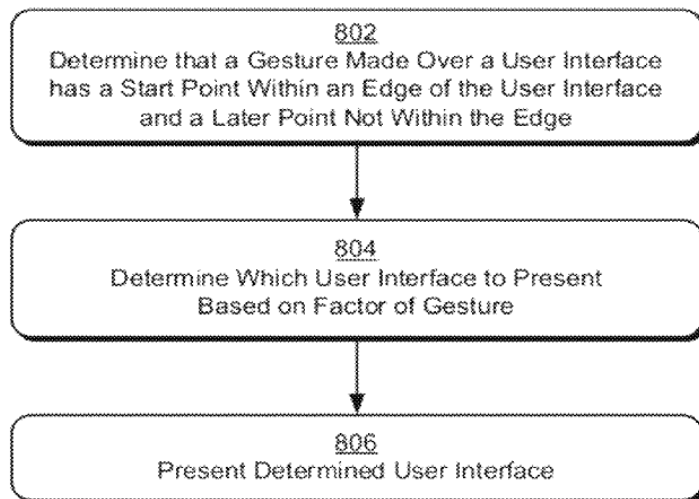
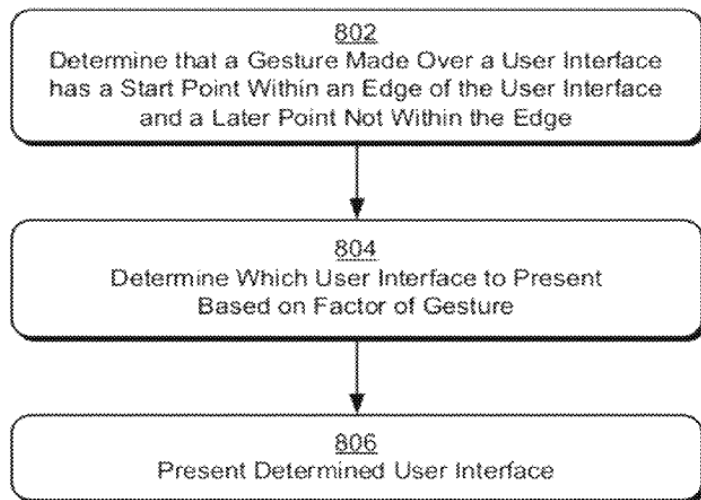


FIG. 8 depicts a method 800 for enabling edge gestures including determining an interface to present based on some factor of the gesture. In portions of the following discussion reference may be made to system 100 of FIG. 1, reference to which is made for example only. Method 800 may act wholly or partly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

Nan Explicitly Teaches Using Gestures with Different Movement Parameters to Switch Between Interfaces

Interface Chosen Based on Gesture Characteristics

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., single-finger or multi-finger), or a speed of the gesture. In some cases, two or more factors of a gesture determine which interface to present such as a drag-and-hold gesture having a drag length and hold time or a drag-and-drop gesture having a drag length and drop position. Thus, block 804 may



Example with “short” and “long” gesture

having a drag length and drop position. 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, speed, or a number of inputs (e.g., fingers).

APPLE-1005, FIG. 8, 7:30-46, 9:9-21
Pet., 42-43, 46-47; Reply, 20-21

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Shiplacoff-Nan Combination Includes Nan's Edge Gestures

Testimony of Dr. Terveen re How Combination Work

34. A POSITA would have been motivated, would have found obvious, and would have predictably and reasonably expected success in beneficially applying the teachings of Nan to Shiplacoff such that Shiplacoff's system can implement gestures such as Nan's edge gestures. Combining the teachings of Shiplacoff and Nan would have merely involved combining prior art elements according to known methods to yield predictable results.

APPLE-1003, ¶34
Pet., 43

Nan

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.

APPLE-1005, 7:9-15
Pet., 46-47

Motivation for Shiplacoff-Nan Combination

Testimony of Dr. Terveen re Motivation

38. Second, Shiplacoff and Nan are directed to improving the transition of displays from one user interface to another. For instance, Shiplacoff teaches transitioning from one mode and corresponding user interface to another, such as the transition from a full-screen mode user interface to a card mode user interface. *See e.g.*, APPLE-1004, FIGS. 1-4. Nan similarly discloses transitioning from one application user interface, such as a webpage interface 306, to an application-specific interface 702. *See e.g.*, APPLE-1004, FIGS. 3, 7, 6:43-60, 4:29-62.

Testimony of Dr. Terveen re Motivation and Predictability

35. In particular, Nan explains that edge gestures are beneficial at least because (1) “edge gestures can be processed before the entire gesture is complete avoiding latency associated with processing entire gestures started elsewhere;” and “valuable real estate used to” display content of user interest, e.g., a movie, “was not taken up with on-screen controls, nor was the user required to remember and find a hardware button.” APPLE-1005, 2:60-3:5. A POSITA interested in avoiding latency and efficient and less disruptive real estate usage would have predictably turned to Nan’s teachings of implementation details to realize these benefits in Shiplacoff’s system.

APPLE-1003, ¶¶38, 35
Pet., 43-45

Shiplacoff Does Not Require Gesture Area

Rebuttal to:

“Petitioner’s Ground 2 further fails because Petitioner has not shown that a POSITA would have sought to modify Shiplacoff to utilize Nan’s ‘edge gesture,’ as doing so would render Shiplacoff’s ‘gesture area’ inoperable.” POR, 60

[0065] For illustrative purposes, device **100** as shown in FIG. **1** is a personal digital assistant or smartphone. Such devices commonly have telephone, email, and text messaging capability, and may perform other functions including, for example, playing music and/or video, surfing the web, running productivity applications, and the like. The present invention can be implemented, according to various embodiments, in any type of device having a touch-sensitive screen, and is not limited to devices having the listed functionality. In addition, the particular layout shown in FIG. **1** is merely exemplary and is not intended to be restrictive of the scope of the claimed invention.

[0066] In various embodiments, touch-sensitive screen **101** and gesture area **102** can be implemented using any technology that is capable of detecting a location of contact and/or user motion proximate to the surface. Alternatively, the present invention can be implemented with other user input mechanisms, such as a keyboard, trackball, stylus, or the like. One skilled in the art will recognize that the techniques of the present invention are well suited to direct manipulation, but that any input technique can be used. In one embodiment,

APPLE-1004, [0065]-[0066]

Pet., 44; Reply, 23-25

Even so, Nan Complements Shiplacoff's Gesture Area

Rebuttal to:

"Petitioner's Ground 2 further fails because Petitioner has not shown that a POSITA would have sought to modify Shiplacoff to utilize Nan's 'edge gesture,' as doing so would render Shiplacoff's 'gesture area' inoperable." POR, 60

Smith Acknowledges Nan's Edge May Be "Off-Screen," Like Shiplacoff's Gesture Area

¶105. That is, Nan teaches that an "edge gesture" is identified by determining whether the starting point of a user input is within an "edge." EX1005, 4:35-5:2; EX2004, ¶105. Thus, in embodiments where the "edge" is off-screen, it is treated the same as an on-screen "edge" for purposes of detecting an "edge gesture." EX1005, 4:35-5:2; EX2004, ¶105.

POR, 62

Nan

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. In some instances, an edge may extend beyond an edge of the display or a screen when an input mechanism is able to receive a gesture portion beyond the display or screen. Example edges are rectangular and vary

APPLE-1005, 4:41-47
Reply, 25

Petition Applied Nan's Edge Gestures to Both [1j] and [1k]

Rebuttal to:

“Notably, Petitioner does not assert that this passage from Nan [i.e., examples provided at 7:30-46] was mapped to Limitations [1k], [15g], and [27f] in the Petition.” Sur-reply, 13

(a) Claim 1 – features [1i]-[1k]

[1i]-[1k]

Under Ground 1, features [1i]-[1k] were addressed, in part, by mapping Shiplacoff's swipe up gesture to the claimed “second input.” To the extent Patent Owner argues that Shiplacoff does not disclose and render obvious a “movement threshold” and related features such as “when the second input is detected to include the movement parameter that is above the movement threshold” (features [1j], [1k]), the edge gesture used in the SNC system (that would substitute the swipe up gesture in Shiplacoff alone) renders these features obvious. APPLE-1003, ¶[87].

As explained above, in SNC, an edge gesture with a certain length, speed, hold time, or a number of inputs, is used to execute a change from the full-screen mode to the card mode. APPLE-1005, FIG. 8, 7:9-55; APPLE-1003, ¶[88]. SNC's system would have determined that an edge gesture occurred when the detected user input movement is above the movement threshold associated with one or more of the length, speed, and number of inputs of the edge gesture. APPLE-1003, ¶[88]. Moreover, consistent with Shiplacoff's swipe up gesture, it would have been obvious and a simple matter of design choice for the edge gesture to include a movement in the upward direction such that there is movement from an edge of the display (as taught by Nan) to a distance across the display (as taught by Shiplacoff). APPLE-1004, ¶[127]; APPLE-1005, FIG. 8, 7:9-55; APPLE-1003, ¶[88].

Pet., 46-47

Issue 3

The Dependent Claims are Similarly Obvious

Dependent Claims 2 and 3

2. The electronic device of claim 1, wherein the electronic device is configured such that the detecting the second input by the second contact that includes movement across the display in the direction, includes detecting the second input by the second contact that includes movement that starts at an edge of the display.

3. The electronic device of claim 1, wherein the electronic device is configured such that the detecting the second input by the second contact that includes movement across the display in the direction, which includes a vertical direction.

Input of [1k], [15g], [27f] Alters “Movement Parameter,” Not Starting Point, of “Swipe-Up”

61. When a gesture different from the swipe up gesture is used, it would also have been obvious to a POSITA that such a gesture would not have the same characteristics and movement parameters as the swipe up gesture so that Shiplacoff's device can distinguish different gestures. For example, it would have been obvious, if not understood, to use a different gesture, such as a tap or flick, that does not have the same movement parameters as those of a swipe up gesture (e.g., touch and slide action for a certain displacement amount and upward direction). Accordingly, it would have been obvious to a POSITA that when a different gesture, such as a tap or flick, is used, such a gesture (“*second input*”) *“does not meet the one or more criteria that is met when the second input is detected to include the movement parameter that is above the movement threshold,”* and would have been returned

APPLE-1003, ¶¶61

Pet. 23-24

By Explaining Starting Point of Swipe-Up Gesture, Petition Addresses Starting Position of Related “Tap” or “Flick”

Shiplacoff (APPLE-1004)

[0127] In another embodiment, the user can switch among the various modes by performing a gesture. One example of such a gesture is a “swipe up” gesture beginning in gesture area 102 and extending onto screen 101. In one embodiment, such a gesture switches from full-screen mode to card mode,

Testimony of Dr. Terveen

in gesture area 102 and extend[s] onto screen 101.” APPLE-1004, ¶[127]. As shown in Shiplacoff’s FIG. 1 below, Shiplacoff’s device would have detected a “swipe up” gesture (e.g., a contact and drag action) (“*detecting the second input by the second contact*”) that began in gesture area 102 and extends onto screen 101. Thus, it would have been obvious to a POSITA that, at least part of the “swipe up” gesture (notably the part that starts at the edge of screen 101) would have included “*movement that starts at an edge of the display.*”

APPLE-1003, ¶62
Pet. 25

Underscored in Nan Combination, Where [1k] Relies on “Edge Gesture”

(i) *Elements [1i]-[1k]:*

87. Under Ground 1, features [1i]-[1k] were addressed, in part, by mapping Shiplacoff’s swipe up gesture to the claimed “second input.” To the extent Patent Owner argues that Shiplacoff does not disclose and render obvious a “movement threshold” and related features such as “when the second input is detected to include the movement parameter that is above the movement threshold” (features [1j], [1k]), the edge gesture used in the SNC system (that would substitute the swipe up gesture in Shiplacoff alone) renders these features obvious.

APPLE-1003, ¶87
Pet. 46-47

Appendix

To Transition to Launcher, Would Have Used Gesture With Different Parameters Than “Swipe Up”

action for a certain displacement amount and upward direction). Accordingly, it would have been obvious to a POSITA that when a different gesture, such as a tap or flick, is used, such a gesture (“*second input*”) “*does not meet the one or more criteria that is met when the second input is detected to include the movement parameter that is above the movement threshold,*” and would have been returned the display, as a matter of design choice, to the launch screen in FIG. 1 that includes the calendar icon (“*replacing the user interface of the application with the home screen including the icon associated with the application*”). APPLE-1004, ¶[99].

APPLE-1003, ¶61;
Pet., 23-24

Touch-Sensitive Area

[0066] In various embodiments, touch-sensitive screen **101** and gesture area **102** can be implemented using any technology that is capable of detecting a location of contact and/or user motion proximate to the surface. Alternatively, the present invention can be implemented with other user input mechanisms, such as a keyboard, trackball, stylus, or the like. One skilled in the art will recognize that the techniques of the present invention are well suited to direct manipulation, but that any input technique can be used. In one embodiment, gesture area **102** provides a touch-sensitive control area extending beyond the edges of touch-sensitive screen **101**, as described in related application Ser. No. 12/115,992, filed May 6, 2008, for “Extended Touch-Sensitive Control Area for Electronic Device”, the disclosure of which is incorporated herein by reference.

APPLE-1004, [0066];
Pet., 15-16, 25

Duarte-992

POR

including manipulating on-screen elements[.]” EX1004, ¶[0076]; EX2004, ¶94. A number of such “conventional” techniques for “direct manipulation of on-screen objects” are disclosed by Shiplacoff through incorporation by reference of Duarte-992. EX1004, ¶[0066]; EX2003, ¶¶[0013], [0006], [0080]; EX2004, ¶94. In

Duarte-992

[0013] In one aspect, the present invention allows some commands to be activated by inputting a gesture within the gesture area. Other commands can be activated by directly manipulating on-screen objects, as in a conventional touch-sensitive screen. Yet other commands can be activated via a combination of these two input mechanisms. Specifically, the user can begin a gesture within the gesture area, and finish it on the screen (or vice versa), or can perform input that involves contemporaneous contact with both the gesture area and the screen. Since both the gesture area and the screen are touch-sensitive, the device is able to interpret input that includes one or both of these areas, and can perform whatever action is appropriate to such input.

SMITH-2003, [0013];
POR, 50

Petition's Additional Discussion of Differentiating Gestures

Petition

APPLE-1003, ¶[80]. In general, it was known in the art that different gestures can be configured by parameters such as a speed, location, time, and direction of touches. APPLE-1008, 19:63-65 (“event information for a swipe event includes two or more of: a location, time stamp, direction, and speed of a swipe”), 18:41-43; APPLE-1009, 18:1-40 (Tables 1 and 2 depicting a time period of 150 ms associated with a swipe gesture); APPLE-1003, ¶[80]. Moreover, a minimum predetermined time period threshold for a touch was also commonly used in the art such that any touch that did not satisfy the minimum predetermined time period threshold would be considered an unintentional touch or another type of gesture, such as a tap gesture. APPLE-1010, 11:2-7; APPLE-1003, ¶[80].

Pet., 37