

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.

Petitioner

v.

SMITH INTERFACE TECHNOLOGIES, LLC

Patent Owner

IPR2024-01088
Patent 10,656,758

PATENT OWNER RESPONSE

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	OVERVIEW OF THE '758 PATENT	5
A.	Multi-Part Pattern-Based Gestures.....	5
B.	Independent Claims 1, 63, and 124	9
III.	LEVEL OF ORDINARY SKILL	11
IV.	CLAIM CONSTRUCTION	12
A.	“Conditional” Limitations	12
B.	“The Duration Of The First Single-Finger Touch Input”	15
C.	Whether Limitations [1.k] and [1.l] Require a Single Continuation Touch.....	17
D.	“User Interface Element”	18
V.	OVERVIEW OF PRIOR ART	19
A.	Chaudhri (EX1004).....	19
B.	Martyn (EX1005)	22
VI.	LIMITATIONS [1.g] / [1.l], [63.c] / [63.h], [124.b] / [124.g]: FAILURE TO CONSISTENTLY IDENTIFY TOUCH ON AND MOVEMENT CRITERIA COMPARED TO “THE FIRST APPLICATION ICON”	24
A.	Failure to Consistently Map “The First Application Icon”.....	24
B.	Failure to Identify Moving “The First Application Icon” Based on “The First Single-Finger Touch Input”	27
1.	<i>Petitioner’s Mapping to Determining A Different Touch Input After “The First Single-Finger Touch Input Concludes” Meets Movement Criteria Does Not Meet the Claim Language.....</i>	27
2.	<i>Petitioner’s Proposed Further Modifications of Chaudhri Rely on Hindsight With No Real Motivation</i>	28

VII. LIMITATIONS [1.j-k], [63.f-g], [124.e-f]: A POSITA IMPLEMENTING MARTYN’S APP SNAPSHOT WOULD NOT ASSIGN A USER GESTURE AS IN THE PURPORTED COMBINATION.....	31
A. Petitioner’s Convoluted Path To The Purported Combination	31
B. Petitioner Fails to Identify A Motivation To Turn To Martyn For A User Gesture To Call The App Snapshot.....	33
C. A POSITA Has No Reason To Turn to Martyn When Chaudhri’s Already Discloses A User Gesture For Calling a Context Menu	38
D. Chaudhri’s Disclosure Assigning The Long Touch Gesture From A Different Function Teaches Away From Assigning That Gesture To Calling The App Snapshot	42
E. Numerous Available User Gestures That Are Consistent, Rather Than Conflict With Chaudhri and Martyn’s Teachings, Further Teach Away From Selecting a Gesture That Creates A “Conflict”	44
1. <i>Correctly adapting Martyn’s “press and hold of the select function when an application name is highlighted” does not match Petitioner’s modification and avoids conflict.....</i>	<i>44</i>
2. <i>Martyn’s other methods of calling up the App Snapshot disclose further options avoiding conflict with Chaudhri</i>	<i>46</i>
F. Petitioner Does Not Identify any Reason or Motivation to Turn to Griffin Beyond a Self-Induced Conflict.....	48
VIII. PETITIONER FAILS TO MEET THE PLAIN LANGUAGE OF MULTIPLE DEPENDENT CLAIMS.....	49
A. Limitations [21] and [22]: Content Indication	49
B. Limitation [35]: Multiple Moved Apps.....	50
C. Limitation [47]: Interactive User Element	51
D. Limitation [57]: Dynamically Generated Content	52
IX. CONCLUSION.....	53

PATENT OWNER'S EXHIBIT LIST

Exhibit	Shorthand	Description
2005	Declaration	Declaration of Vijay Madisetti, Ph.D.
2006	CV	CV of Vijay Madisetti, Ph.D.
2007	Deposition	Deposition of Andrew Cockburn, Ph.D.
2008	'918 Provisional	Provisional Patent Application 61/581,918

I. INTRODUCTION

Petitioner challenges the three independent claims of the '758 Patent: Claims 1, 63, and 124 (the “Challenged Independent Claims”). The Challenged Independent Claims each recite “detecting a first single finger touch input” “at a location” “that corresponds to a first application icon” and then a series of limitations that collectively recite logic for detecting, on a touch screen, aspects of multi-part, multi-action pattern-based user gestures and performing various functions “in accordance with” / “based on” “a determination” “that the first single-finger touch” meets various criteria. Each Challenged Independent Claim includes four limitations each reciting a separate “in accordance with” / “based on” and function pair, all of which must be included in the device logic to meet the claim. Limitations [1.i-l], [63.e-h], [124.d-g]. Petitioner’s mapping fails for multiple Limitations.

First, Limitations [1.g], [63.c], [124.b] recite “detect[ing] a first single-finger touch input” “at a location on the touch-sensitive surface that corresponds to a first application icon.” Limitations [1.l], [63.h], [124.g] then recite “moving *the* first application icon,” which must, under the plain meaning, be the “*the* first application icon” mapped for Limitations [1.g], [63.c], [124.b]. Yet, in violation of the plain language, for Limitations [1.l], [63.h], [124.g], Petitioner points only to moving a *different* application icon, not “*the* first application icon” mapped for Limitations [1.g], [63.c], [124.b].

Further, Limitations [1.l], [63.h], [124.g] recite performing the “moving” function “in accordance with” / “based on” “a determination that *the* first single-finger touch input meets the one or more movement criteria.” Again in violation of the plain language, Petitioner points only to a determination evaluating a later, different touch after “*the* first single-finger touch input” has concluded. Petitioner proposes two obviousness modification to attempt to overcome this flaw, but both start from the premise that Petitioner is free to select “design options to implement Chaudhri’s teachings,” which assumes Chaudhri is silent on a particular implementation. Pet., 38. That premise fails because Chaudhri explicitly states that the two touches Petitioner maps are two distinct, discontinuous touches—“contact 5008” and “contact 5016-a.” EX1004, [0217], [0220].

Second, Limitations [1.j-k], [63.f-g], [124.e-f] recite performing separate functions “in accordance with” / “based on” “a determination that the first single-finger touch input meets the one or more movement criteria” (a) “the duration of the first single-finger touch input is evaluated to be greater than the first time threshold,” and (b) “the duration of the first single-finger touch input is evaluated to be greater than a second time threshold that is greater than the first time threshold.” Petitioner colloquially refers to these criteria as a “hold” and “longer hold.” Pet., 1.

Petitioner does not assert that Chaudhri discloses evaluating any touch on an application icon for both a “hold” and “longer hold” or that Chaudhri discloses the

function required by Limitations [1.k], [63.g], [124.f]. Pet., 10-11. Rather, Chaudhri discloses evaluating the duration of a user's touch for exceeding a single "predetermined period of time" to call a single function: "enter[ing] a user interface reconfiguration mode." *Id.* Thus, to reach the claims, Petitioner must, through modifications: (a) add a new function: Martyn's "App Snapshot," (b) assign a hold gesture (already occupied by a different function) to calling that App Snapshot, and (c) assign a longer "hold" to entering reconfiguration mode. Petitioner reaches this goal by: (a) purporting to adopt a context menu calling method ("a press and hold of the select function when an application name is highlighted") from Martyn, (b) changing that calling method to a completely different method (a touch and hold on an application icon) through a single conclusory sentence labeling this different method to be "analog" of what Martyn discloses; (c) declaring that it had created a "**problem** of distinguishing between different gesture commands" by assigning two functions to the same touch and hold gesture, and (d) purporting to solve that "**problem**" by assigning one gesture the "longer hold" and the other the "hold," not coincidentally choosing the order matching the claims. Pet., 15-22.

Petitioner's convoluted path includes multiple fatal flaws. First, while Petitioner purports to disclose four motivations to turn to Martyn, all four motivations address only why a POSITA would turn to Martyn's disclosure of an App Snapshot contextual menu, not why a POSITA would turn to Martyn for a

method of calling a context menu, let alone the particular “press and hold of the select function when an application name is highlighted” method Petitioner chooses. Second, Petitioner provides no credible basis to change from Martyn’s disclosed gesture to a touch and hold on an application icon. Chaudhri discloses using a click wheel to scroll a highlight through icons and call a function, providing a direct basis to implement Martyn’s teachings that does not match Petitioner’s combination (or the claims). Third, Petitioner touts that Chaudhri already discloses, in a different mode, using a context menu (Pet., 18-19), but fails to acknowledge that this disclosure includes a calling method—tapping on a target region in the corner of the icon—that is readily applicable to calling Martyn’s App Snapshot in “normal mode.” This leave a POSITA no reason to go beyond Chaudhri and consult Martyn for a method of calling the App Snapshot.

Fourth, Chaudhri’s assignment of a touch and hold gesture on an application icon to a different function (entering reconfiguration mode) *teaches away* from assigning a second function to that gesture because, as Petitioner acknowledges, it creates a “*problem* of distinguishing between different gesture commands.” Pet., 21. While in *KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 425 (2007), the Court permitted a first modification that led to a second problem, the facts are fundamentally different here. In *KSR* where a first modification caused a second problem (“wire-chafing”), there was *a reason* to make the first modification (“the

objective to upgrade Asano to work with a computer-controlled throttle”). Here, Petitioner has identified no reason to adopt a touch and hold gesture. Rather, Petitioner’s arrival at that conflict is guided only by hindsight, unsupported by any motivations and deterred by Chaudhri as a whole.

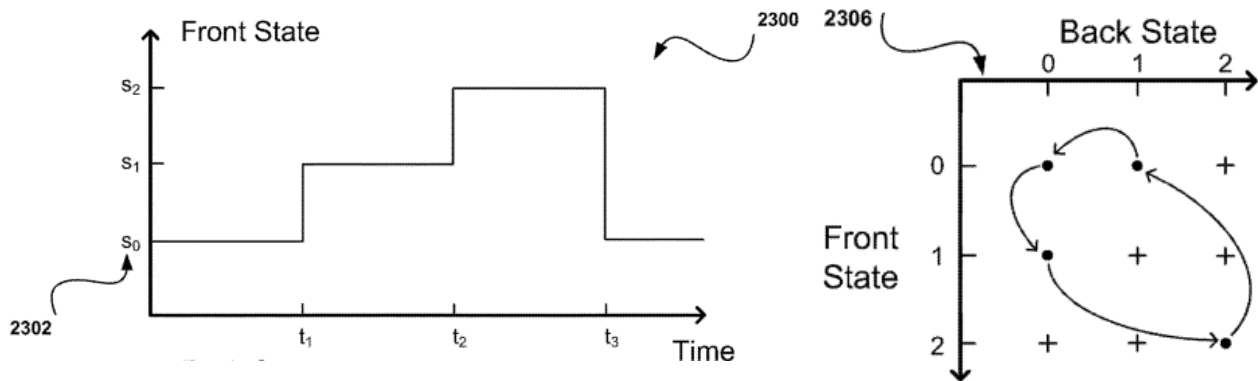
A single one of the failures, in either sets of limitations, discussed above is sufficient to reject the mapping of all Challenged Claims. Additionally, Petitioner’s mappings of multiple dependent claims include additional flaws. Accordingly, all Challenged Claims should be found not unpatentable.

II. OVERVIEW OF THE ’758 PATENT

A. Multi-Part Pattern-Based Gestures

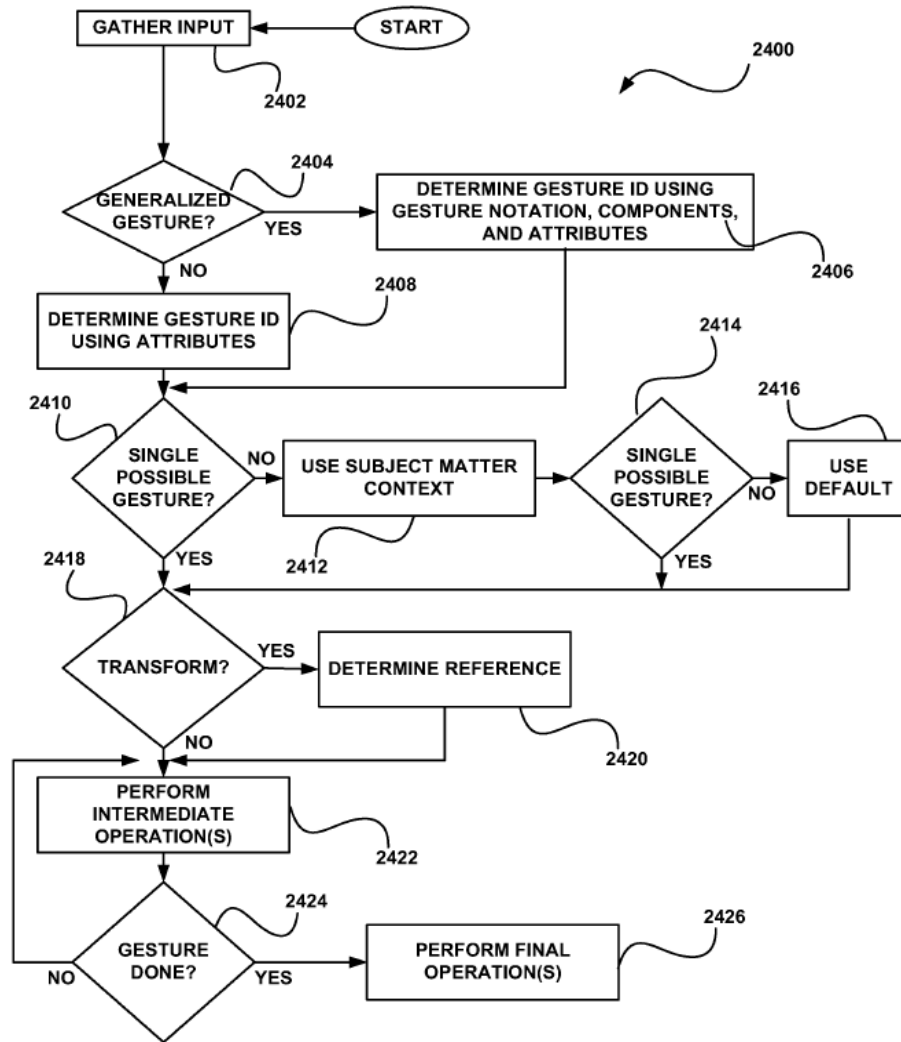
At the time of the ’758 Patent’s December 2011 priority date, touch screen mobile devices were becoming increasingly capable of more complex tasks that required more control than possible with basic input methods. EX2005, ¶¶28-30 (describing examples of mobile games and remote work tasks). Further, limited screen real estate meant that input and output techniques used for more complex tasks on desktop computers could not be used. *Id.* The ’758 Patent addresses this shortcoming by enabling users to operate these new powerful mobile devices and their feature-rich applications through implementation of multi-part gestures where users can take successive actions and receive feedback at each stage. *Id.*

The '758 Patent describes these multi-part, multi-action gestures as “pattern-based gestures” that “may be recognized in the form of an ordered sequence” by, for example, “using a decision tree.” EX1001, 75:37-39; EX2008, [00581]. The “pattern-based gesture” may be defined as “a path through a phase space of one or more dimensions.” EX1001, 73:39-41; EX2008, [00564]. Exemplary “pattern-based gesture 2302” is “described by a phase space path which represents the time evolution of the touch state associated with a contact point” on the touch screen. EX1001, 73:53-56; EX2008, [00566].



EX1001, FIG. 23; EX2008, FIG. 28. Each state, and transitions between states, may be defined by factors including “contact point changes” and “the time spent on each leg of the path through phase space.” EX1001, 73:59-74:19; EX2008, [00567]-[00570]. Consistent with the “decision tree” aspect of these pattern-based gestures, multiple patterns may share the same starting (and intermediate) points. EX1001, 75:37-39; EX2008, [00581]; EX2005, ¶31 (explaining how decision tree logic conforms to a user’s expectations).

To detect these pattern-based gestures, the '918 Provisional, which the '758 Patent incorporates by reference (EX1001, 1:24-26), discloses “gesture differentiation,” which is distinguished from simply detecting a “basic user interaction (i.e., button press, basic application interaction, etc.),” techniques. EX2008, [00498], [00502]. The process begins by “determine[ing] whether there is any meaningful user input,” including applying a “threshold above which user interaction may rise before it may be deemed meaningful.” *Id.*, [00501]-[00502]. Detecting the gesture may involve “gesture subject matter context,” including “objects displayed at or near the location of the gesture.” *Id.*, [00512]. For example, the starting point may be the beginning of a touch contact at a particular location of a specific object on the touch screen. *Id.*; EX2005, ¶32.



EX2008, FIG. 24.

Again, as indicated by the chart, multiple full patterns may share the same starting (and intermediate) points. EX2005, ¶33 (explaining diagram). Consistent with the multi-part nature, the device may perform “intermediate operations associated with the identified gesture.” EX2008, [00521]. “[E]ach execution of intermediate gesture operations may trigger a display update (i.e. rendering, etc.).” *Id.*, [00524]. One purpose of these intermediate gestures is so “a user may receive feedback on their gesture before terminating the interaction.” *Id.*, [00522].

A “gesture may be ended by ending one or more user inputs” or “by performing a different gesture” at which time “the final operations associated with the identified gesture are performed.” *Id.*, [00526]-[00528]. “For example, in one embodiment, a gesture which triggers the display of a pop-up menu and allows a user to cycle through menu items may have as a final operation the selection of the current menu item and removal of the pop-up menu.” *Id.*, [00529].

B. Independent Claims 1, 63, and 124

Petitioner challenges the three independent claims of the ’758 Patent: Claims 1, 63, and 124 (“Challenged Independent Claims”). Claim 1 is directed to an “electronic device.” EX1001, 97:29. Claim 63 is directed to a method. EX1001, 108:41. Claim 124 is directed to a “non-transitory computer readable storage medium.” EX1001, 114:60. Claims 1, 63, and 124 are substantially similar.

The Challenged Independent Claims recite logic to process user inputs on a touch screen to recognize a set of gesture components that begin from a common starting point and perform operations and/or provide feedback upon recognizing various aspects of a gesture. EX2005, ¶36-38. Claims 1, 63, and 124 each recite “an application launching user interface that includes a plurality of application icons,” detecting a first single-finger touch input” “at a location on the touch-sensitive surface that corresponds to a[n] ... application icon,” and “determining a response based on evaluating the” “duration” or “movement” of the “touch input

against” various criteria. Limitations [1.f-g], [63.b-d], [124.a-c]. Each claim then recites four logic operations:

- “[in accordance with] / [based on] a determination that ... ***the duration of the first single-finger touch input is evaluated to be less than a first time threshold***, launching the first application,”
- “[in accordance with] / [based on] a determination that the first single-finger touch input ... ***the duration of the first single-finger touch input is evaluated to be greater than the first time threshold***, displaying one or more action option objects associated with the first application without launching the first application,”
- “[in accordance with] / [based on] a determination that the first single-finger touch input ... ***the duration of the first single-finger touch input is evaluated to be greater than a second time threshold that is greater than the first time threshold***, performing an operation ... , ***and***”
- “[in accordance with] / [based on] a determination that ***the first single-finger touch input meets the one or more movement criteria***, moving the first application icon in a foreground virtual display layer so that the first application icon appears to float above a background virtual display layer.”

Limitations [1.i-l], [63.e-h], [124.d-g].

III. LEVEL OF ORDINARY SKILL

PO agrees with the level of ordinary skill in the art adopted by the Board for the purposes of the Institution Decision. *See* Paper 7, 8-9. Specifically, a POSITA:

“would have had ‘a bachelor’s degree in computer science, computer engineering, or a comparable field and at least two years of professional experience working with graphical user interfaces and applications for computers.’”

Id., 8.

PO agrees with the Board’s rejection of two other aspects of Petitioner’s proposal. *Id.*, 8-9. First, the phrase “[a]dditional years of experience could substitute for an advanced-level degree (and vice versa)” (Pet., 7) is inapplicable where the base degree is a bachelor’s degree as here and was likely a simple copy and paste error. Dr. Cockburn confirmed that deleting “additional years of experience could substitute for the advanced-level degree” from his definition cited in the Petition would make “more sense.” *See* EX2007, 9:7-23.

Second, PO agrees with the Board that Petitioner’s additional statement that a POSITA “would have possessed extensive background knowledge and understanding of the pertinent technologies” listing specific areas (Pet., 7) should not be adopted. *See* Paper 7, 9. This additional requirement is either redundant with the level of ordinary skill articulated above, or inconsistent to the extent it seeks to impose a level of ordinary skill beyond “a bachelor’s degree ... and at least two years

of professional experience.” EX2005, ¶27. Further, to the extent inclusion of these statements permits Petitioner to pivot to references the Petition does not identify for a given ground and element, that inclusion is contrary to the requirement that the Petition include an “identification of particular portions of prior art on which the petitioner is relying.” *Corephotonics, Ltd. v. Apple Inc.*, 84 F.4th 990, 1001 (Fed. Cir. 2023).

IV. CLAIM CONSTRUCTION

A. “Conditional” Limitations¹

Limitations [1.i-l], [63.e-h], [124.d-g] collectively recite four actions to be performed “[in accordance with] / [based on]” a “determination that the first single-finger touch meets” four recited “criteria.” These four Limitations collectively describe logical functions/steps to process instances of user input on a touch screen that begin from a common starting point—the start of “the first single-finger touch input.” EX2005, ¶39. This is consistent with “decision-tree” and detection logic for detecting multi-part pattern-based gestures as described in connection with Figure 23 of the ’758 Patent and Figure 24 of the incorporated-by-reference ’918

¹ PO uses the term “conditional” limitation to refer to limitations that recite an action to be performed when a precondition is met without conceding such form is sufficient to meet any particular case law.

Provisional as further discussed in Section II.A, above. EX2005, ¶40 (describing the specific word choice as a particular example of decision tree logic).²

Limitation [1.i] will not be met during the same instance of a user touch as Limitations [1.j] or [1.k] (and vice versa), but that is not a basis, as a matter of law, to relieve Petitioner from demonstrating the alleged prior art combination includes **logic** meeting each Limitation such that each limitation will be performed when presented with an instance of user input “where their preconditions apply.” See *Seirra Wireless, ULC v. Sisvel S.P.A.*, 130 F.4th 1019, 2025 U.S. App. LEXIS 5458, *5 (Fed. Cir. Mar. 10, 2025). As an initial matter, *Ex Parte Schulhauser*, No. APPEAL 2013-007847, 2016 WL 6277792 (P.T.A.B. Apr. 28, 2016) (precedential), at best, applies to Claim 63, which is a method claim. “[A] **system claim** [reciting]

² While PO has demonstrated that the plain language of the claims is consistent with the specification disclosure, PO respectfully notes that it was not incumbent on PO to do so nor would the lack of any specific embodiment be a justification to ignore the plain language. As the Federal Circuit recently reaffirmed, “the patentee is free to choose a term and expect to obtain the full scope of its plain and ordinary meaning unless the patentee explicitly redefines the term or disavows its full scope.” *Iqris Techs. LLC v. Point Blank Enters., Inc.*, 130 F.4th 998, 2025 U.S. App. LEXIS 5316, *13 (Fed. Cir. Mar. 7, 2025).

structure that performs a function, which only needs to occur if a condition precedent is met, *still requires structure for performing the function should the condition occur.*” *Id.*, *7. (emphasis added); *see also See Ex Parte John Rebstock & Christopher K. Schrichte*, No. APPEAL 2022-003164, 2023 WL 4643799, at *4 (P.T.A.B. June 23, 2023)(“We construe the scope of the computer program product claim 17, as being such that the computer program has the capability of performing operations of both the alternatives”).

As to Claim 63, nothing in the plain language suggests any of the recited conditional Limitations are optional or may be ignored in mapping the prior art. Rather, “the plain and unambiguous language” of the claims requires that “to come within the claim” each element must be performed “where their preconditions apply.” *See Seirra Wireless, ULC*, 2025 U.S. App. LEXIS 5458 at *5. In *Seirra Wireless* the Federal Circuit emphasized the “and” between two method steps that the Board had deemed “cannot both occur in response to the same set of stimuli” indicated each step must be performed “where their preconditions apply.” *Id.* at *4-*5. Like in *Sierra Wireless*, the claims here link the conditional limitations with “and” indicating each Limitation must be performed where their preconditions apply. Limitations [1.k], [63.g], [124.f]. Further, Claim 63 more closely resembles the claim the Federal Circuit addressed in *Hytera Communications Co. v. Motorola Solutions, Inc.*, 841 F. App’x 210 (Fed. Cir. 2021), which distinguished *Schulhauser*.

There, the Federal Circuit found the method claim at issue “requires that a specific action be taken in response to each of two alternative specified conditions,” and the claim “is not met unless the TDMA system is configured to perform each claimed responsive action.” *Id.*, 216; *see also Ex Parte Brosius*, Appeal No. 2010-001734, 4-5 (BPAI January 19, 2012) (citing to *Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363 at 1372 (Fed. Cir. 2003) (holding that two conditional method “limitations (b) and (c) are not optional because if their condition is satisfied, these limitations *require* certain steps to be performed”) (emphasis in original)).

The same is true here. Claim 63 includes multiple conditional steps that specify different actions based on the evaluation of the touch input against various criteria. EX2005, ¶41. Each condition leads to a different action, and understanding the claim fully requires evaluating all possible conditions and their corresponding actions. *Id.* Further, like the recitation of “in a TDMA system” in the claims analyzed in *Hytera*, claim 63 recites the context of the method being performed “at an electronic device including a display and a touch-sensitive surface” (Limitation [63.a]), which, as discussed above, connotes a group of logic all expected to be met when a given precondition applies. *Id.*

B. “The Duration Of The First Single-Finger Touch Input”

Each Challenged Independent Claim recites “detecting a first single-finger touch input” and then recites multiple limitations each referencing different

“determination[s] that the first single-finger touch input meets” a criteria, including three limitations that reference “the duration of the first single-finger touch input.” Limitations [1.g-1], [63.c-h], [124.b-g]. The claims do so in the context of describing programming logic that will receive and act upon numerous instances of user input in accordance with that logic. *See* Limitations [1.e] (“the one or more programs including instructions for:”), [63.a] (“at an electronic device”), [124.pre] (“the one or more programs comprising instructions which ... cause the electronic device to”). EX2005, ¶42.

With this context, the antecedent bases “the duration of” and “the first single-finger touch input” indicate that evaluating the preconditions recited in each Limitation requires evaluating instances of user input beginning at a common starting point. Limitations [1.g], [63.c], [124.b]; EX2005, ¶¶28-34, 43. Here, the plain language of the claims defines that common starting point as the initial detection of a “single finger touch input at a location ... that corresponds to a first application icon.” Limitations [1.g], [63.c], [124.b]; EX2005, ¶44. Accordingly, it is not sufficient, in collectively mapping the conditional Limitations, to point to logic in the prior art for detection of gestures that do not share the common starting point or that are determined based on “durations” starting at any point other than the common starting point. EX2005, ¶¶44-45.

At the same time, because the claims refer to the logic used to evaluate numerous instances of user input, it is not necessary that the prior art include logic where the preconditions for each Limitation would be met in response to a *single* user input. EX2005, ¶46. This is again consistent with “decision-tree” and detection logic for detecting multi-part pattern-based gestures as described in connection with Figure 23 of the ’758 Patent and Figure 24 of the incorporated-by-reference ’918 Provisional as further discussed in Section II.A, above. EX2005, ¶47. It is also consistent with the fact that some, but not all, of the Limitations recite mutually-exclusive preconditions such that requiring all preconditions to be met in response to a single instance of user input is contrary to the plain claim language. EX2005, ¶¶48-49 (describing the mutual exclusivity of Limitations [1.i] and [1.j]).

C. Whether Limitations [1.k] and [1.l] Require a Single Continuation Touch

Petitioner identifies a claim scope issue, but does not take a position as to whether the Challenged Claims should be “interpreted to require a single continuous touch between Elements [1.k] and [1.l].” Pet., 38. As an initial matter, the ’758 Patent’s disclosure of multi-part gestures includes multi-part gestures that do **not** involve a single continuation touch across all parts. See Section II.A, above; see EX2008, [00529] (describing a multi-part “gesture” that “triggers the display of a pop-up menu,” “allows a user to cycle through menu items,” and “may have as a final operation the selection of the current menu item and removal of the pop-up

menu”). However, the specific claim language of the Challenged Claims requires a continuous touch. EX2005, ¶51. Specifically, Limitation [1.g] recites “detection a first single-finger touch input” “at a location ... that corresponds to a first application icon,” and subsequent Limitations [1.k] and [1.l] each refer to “the first single finger touch.” Limitation [1.k] further refers to “the duration of the first single finger touch,” referring to the elapsed time since the initial touch detected in Limitation [1.g]. EX2005, ¶52. The context of the Challenged Claims, including the use of the word “touch” and identifying the start and duration of that touch, as well as the use of “the first single finger touch” in each limitation connote to a POSITA that each refers to a touch that has not concluded on lift-off. *Id.* The language in the Challenged Claims assessing movement criteria for “the first single finger touch” stands in contrast to the language of other family members, such as Claim 1 of U.S. Patent No. 10,671,212, which recites “a touch” and “the touch” for several claim elements, but then recites separate subsequent detection of “a slide or swipe gesture.” Thus, the Challenged claims could have referred to “a” different gesture to refer to a different touch but they do not. EX2005, ¶53-54.

D. “User Interface Element”

Claim 47 requires that a “user interface element is displayed to appear at least partially behind the first application icon.” A POSITA would understand that the plain and ordinary meaning of a “user *interface* element” is an element of a device

that a user *interacts* with, which requires more than simply being a “*display*” element. EX2005, ¶55. The specification of the ’758 Patent is consistent with this plain meaning. For example, the specification discloses associating a “*context determination zone*,” which “refers to the area that is examined while determining the context of a selection,” with a corresponding “*user interface element*[.]” EX1001, 28:43-44, 28:60-67, 29:27-36; *see also id.*, 29:27-32 (a “determination” of a “contact point” related to a “context determination zone” of a “UI control widget” “may be performed using the *bounding boxes associated with user interface elements*”); EX2005, ¶56. As another example, “[i]n various embodiments, a tipping pressure gesture may be used to interact with user interface elements.” EX1001, 66:3-13. Finally, the specification explains that “a *display* element refers to the region of the display allotted to an identifiable visual object (*id.*, 70:38-40), which is broader than “user *interface* element,” consistent with the plain meaning of “interface” requiring more than just “display.” EX2005, ¶57-58.

V. OVERVIEW OF PRIOR ART

A. Chaudhri (EX1004)

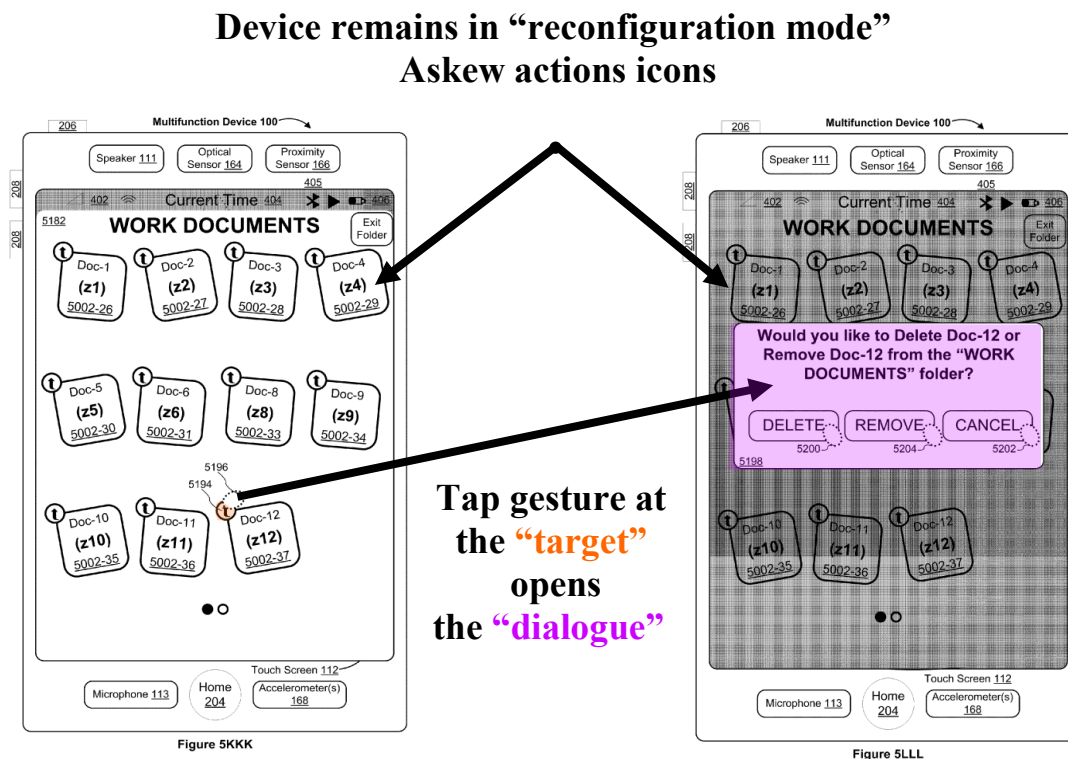
Chaudhri describes “electronic devices with touch-sensitive surfaces, including but not limited to electronic devices with touch-sensitive surfaces for managing folders.” EX1004, [0003]. Chaudhri’s “device provides the functionality of responding” to many input methods. EX1003, ¶75; EX2005, ¶60; EX1004,

[0085], [0087], [0093], [0095], [0097], [0101]-[0102], [0152], [0164], [0173]-[0174], [0211], [0218], [0380]. Chaudhri describes “input control devices [such as] physical buttons (e.g. push buttons, rocker buttons, etc.),” a “physical or virtual wheel (e.g. a click wheel) as an input control device,” an “optical sensor controller 158 in [an] I/O subsystem,” an “accelerometer 168,” and “microphone 113” for “verbal input.” EX1004, [0085], [0093], [0095], [0097], [0176]; EX2005, ¶¶61-62.

Chaudhri also discloses a “normal operation mode” and a “user interface reconfiguration mode.” EX1004, [0217-0218]; EX2005, ¶¶63. In the “normal mode, tapping on an application will launch the application.” EX1004, [0217]; EX2005, ¶¶63. In the “user interface reconfiguration mode,” a user can rearrange applications and can reach a “pop-up dialogue” by touching a separate “object modification target.” EX1004, [0274]; EX2005, ¶¶63. Petitioner does not assert and Chaudhri does not disclose reaching a context menu via a durational touch. Pet., 15-22.

While in “user interface reconfiguration mode,” “an object removal badge (e.g., 5010 in FIG. 5B)” appears and “when the device detects activation of the object removal badge (e.g., tap gesture 5011 in FIG. 5B).” EX1004, [0218]. “In some embodiments, instead of, or in addition to, the selectable user interface object removal region described above, the device displays an object modification target associated with one or more of the selectable user interface objects.” EX1004, [0273]. This object modification target is displayed while in “user interface

reconfiguration mode.” EX1004, [0274]. While in this mode the device may “detect[] a request to activate the [orange] object modification target [5194] for a respective selectable user interface object (e.g., tap gesture 5196 at a location on the touch screen 112 that corresponds to a location of an object modification region for action icon 5002-37), the device displays an object modification dialogue.” EX1004, [0274]. “For example in FIG. 5LLL the device displays a [purple] pop-up dialogue 5198 that provides the user with options for modifying the selectable user interface object.” *Id.* The Petition describes this resulting pop-up dialogue as “displaying a contextual ‘pop-up dialogue’ sub-menu that ‘provides the user with options for modifying the selectable user interface object.’” Pet., 18.



EX1004, Figures 5KKK-LLL (annotated).

B. Martyn (EX1005)

Martyn discloses an “App Snapshot”³ for “small screen devices.” EX1005, [0009], EX2005, ¶65-66. The summary window displays “a limited list of common functions and commonly accessed stored data” that “saves the user from navigating to the required application, opening it up, and then navigating within that application to enable the data of interest to be seen.” EX1005, [0009]-[0015].

Petitioner relies on Martyn as supplementing Chaudhri’s disclosure of a touch screen app launcher not just “App Snapshot” functionality, but for the user gesture calling the App Snapshot. Pet., 16, 32-33. But Martyn does not mention touch screens and, among at least six methods Martyn discloses, none relate to any sort of touch on a portion of the display representing an application for which the App Snapshot is desired. EX1005, [0016]-[0017]; EX2005, ¶65; EX2007, 28:14-25.

The first method Martyn discloses is a *two-step* process of, first, navigating a highlight to a desired application by “tak[ing] conventional navigation steps, such as scrolling, to move the highlight” and, second, “call[ing] or fir[ing]” the Snapshot by ceasing scrolling such that “the highlight rest[s] on the name of an application in

³ Martyn uses several different terms interchangeably: “App Snapshot,” “summary window,” and “snapshot view.” EX1005, [0009-0010], [0015], [0017-0026], [0028-0031].

the App Launcher for a certain amount of time” until a “timeout” is reached.

EX1005, [0016]-[0017]; EX2005, ¶67, 72.

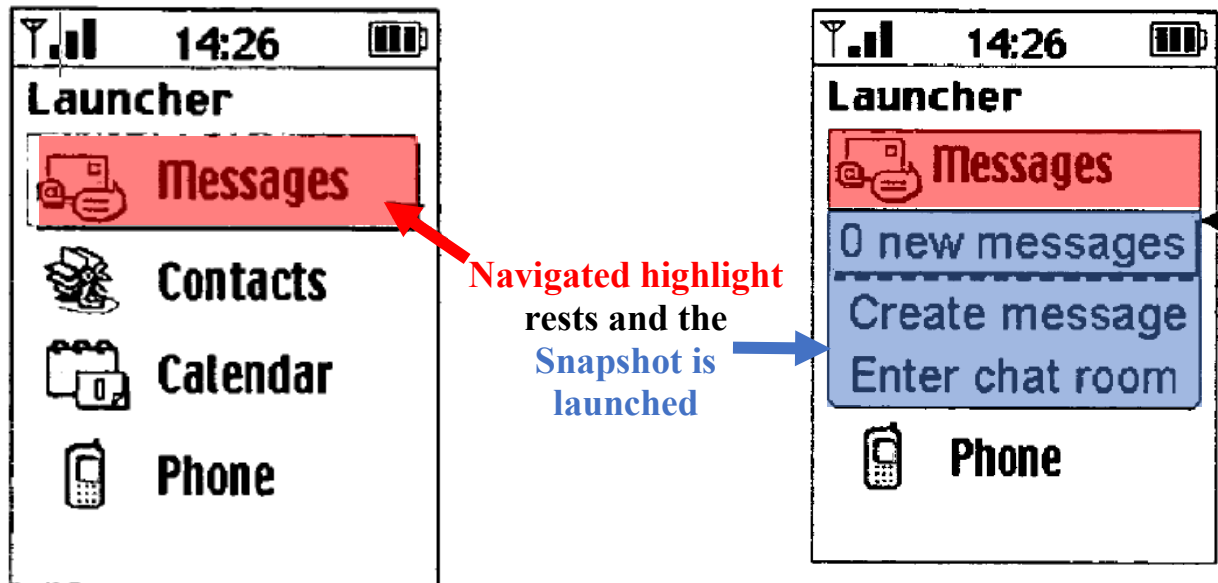


Figure 1

Figure 2

EX1005, FIGs. 1, 2 (annotated).

Martyn lists five more “other” or “example” selection processes: “voice activation,” “softkey selection,” “using a right scroll function (if the mobile telephone has 4-way Scrolling capacity),” “a press and hold of the Select function when an application name is highlighted,” and “or using a right cursor key on a highlighted application name.” EX1005, [0017], EX2005, ¶68-69. Except for “voice activation,” which could be a single voice command identifying the application and desire to launch the App Snapshot (EX1007, 34:2-35:12), each of these additional techniques are similarly two-part processes that first require

scrolling a highlight to an application and then performing the second action. EX2005, ¶70. Most relevant here, the method Petitioner relies upon explicitly states “a press and hold of the Select function *when an application name is highlighted*,” indicating the required first step of navigating to the application. EX1005, [0017]; EX2005, ¶71; EX2007, 39:3-12.

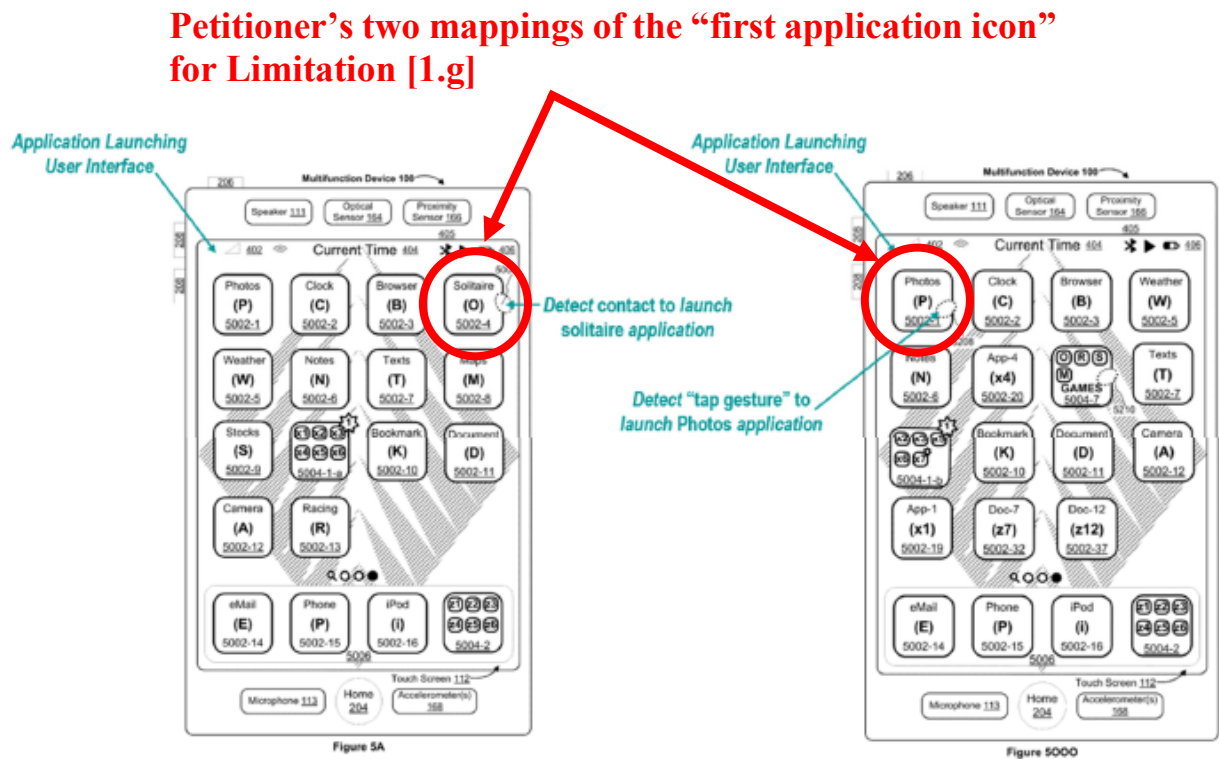
VI. LIMITATIONS [1.g] / [1.l], [63.c] / [63.h], [124.b] / [124.g]: FAILURE TO CONSISTENTLY IDENTIFY TOUCH ON AND MOVEMENT CRITERIA COMPARED TO “THE FIRST APPLICATION ICON”

A. Failure to Consistently Map “The First Application Icon”

Limitations [1.g], [63.c], [124.b] recite “detect[ing] a first single-finger touch input” “at a location on the touch-sensitive surface that corresponds to a first application icon.” Limitations [1.l], [63.h], [124.g] then recite “moving the first application icon,” which must, under the plain meaning, be the “the first application icon” mapped for Limitations [1.g], [63.c], [124.b].

Petitioner provides two alternative mappings of Limitations [1.g], [63.c], [124.b], mapping the “first application icon” to either: (a) the Solitaire icon or (b) photos application icon. Pet., 26-27, 61. Petitioner does so first through prose, parentheticals, and emphasis: (a) “Chaudhri explains that ‘the device *detects* a contact 5008 **with the solitaire application icon**’ and performs a responsive action (e.g., *launching the application* or entering the reconfiguration mode) based on the characteristics of the touch;” (b) “as shown in Figure 5000, ‘*detecting* a tap gesture

5208 at a location on the touch-sensitive surface that corresponds to the **photos action icon** 5002-1' causes the device to '*launch[]* the **photos application**.'" *Id.*, 26 (bold italics emphasis in original, highlighting added). Petitioner affirms these two mappings through arrows and text annotating Figures 5A and 5000:



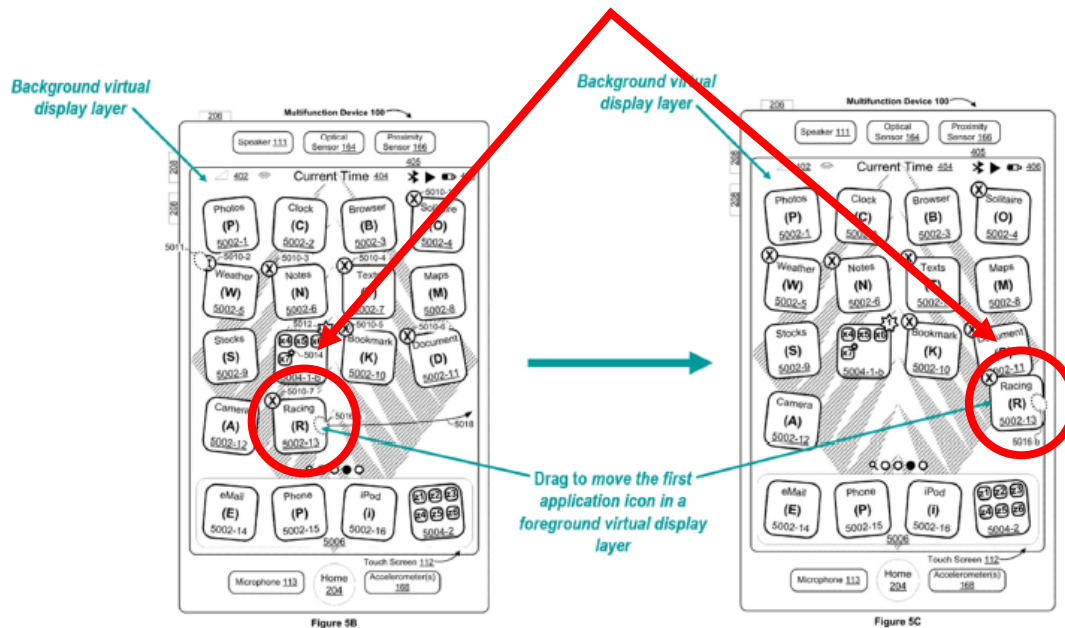
EX1004, Figures 5A (annotated), 5000 (annotated); EX1003, ¶85.

Pet., 26.

For Limitations [1.l], [63.h], [124.g], Petitioner does not identify “moving” either the Solitaire or Photos icon. Pet., 36-38, 61. Instead, Petitioner states that the device “detects a touch ‘at a location that corresponds to a **“racing” [application] icon**’ coupled with ‘subsequent movement’ of the touch ‘to the edge of the touch-

sensitive surface,’ the *application icon* is likewise ‘*moved*’ to the edge of the display.” *Id.* (bold italics emphasis in original, highlighting added). Petitioner affirms this mapping through arrows and text annotating Figures 5B-5C:

Petitioner’s mapping of the “first application icon” for Limitation [1.]



EX1004, Figures 5B-C (annotated); EX1003, ¶100.

Pet., 36.

Having previously established the “first application icon” as either the Photos icon or the Solitaire icon, Petitioner’s mapping to Chaudhri’s disclosure of moving the Racing icon is insufficient to meet the plain language of the Limitations [1.1], [63.h], [124.g] of moving “**the** first application icon.” Petitioner provides no

alternative mappings that could meet the plain language. Pet., 36-38. Thus, Petitioner fails to map Limitations [1.l], [63.h], [124.g].

B. Failure to Identify Moving “The First Application Icon” Based on “The First Single-Finger Touch Input”

1. *Petitioner’s Mapping to Determining A Different Touch Input After “The First Single-Finger Touch Input Concludes” Meets Movement Criteria Does Not Meet the Claim Language*

As discussed in Section IV.C, above, the plain language of Limitations [1.l], [63.h], [124.g] “a determination that the first single-finger touch input meets the one or more movement criteria” requires determining a portion the touch that began with the initial touch detected in Limitations [1.g], [63.c], [124.b] meets the movement criteria. EX2005, ¶77-78. And, because “the first single-finger touch input” concludes upon lift off, determining that a different touch occurring after “the first single-finger touch input” concludes meets the movement criteria is insufficient to meet the plain claim language. *Id.*

Petitioner does not contend that its primary mapping meets the claim requirement. Pet., 35-38. Nor can it. Chaudhri discloses “the device detect[ing] a *contact 5008* with the solitaire application icon 5002-4 for more than a predetermined period of time (e.g., 2 seconds) and, in response, the device enters a user interface reconfiguration mode.” EX1004, [0217]; Pet., 33-34 (citing same for Limitation [1.k]). Then, separately, *after* the device is in reconfiguration mode, the device detects a *separate “contact 5016-a* on the touch-sensitive surface (e.g., touch

screen 112) at a location that corresponds to a “racing” action icon 5002-13, and subsequent movement 5018 of the contact to the edge of the touch-sensitive surface.” EX1004, [0220]; Pet., 36 (citing same for Limitation [1.1]). A POSITA would understand that the reference to two distinct contacts and separately detecting those contacts refers to two separate discontinuous touches. EX2005, ¶80-81. Chaudhri’s reference to “subsequent movement” of “contact 5016-a” without identifying a new contact further confirms that the reference to two separate contacts refers to two separate discontinuous touches. *Id.* Accordingly, Petitioner’s mapping to Chaudhri’s disclosure does not meet Limitations [1.1], [63.h], [124.g].

2. *Petitioner’s Proposed Further Modifications of Chaudhri Rely on Hindsight With No Real Motivation*

Aware that its primary mapping does not meet the plain language requirements of Limitations [1.1], [63.h], [124.g], Petitioner proposes two alternative obviousness modifications that contradict with each other and are both founded upon a mischaracterization of Chaudhri to create a reason to seek modification that does not exist. Pet., 38-39, 61-67. Petitioner asserts that “[a] POSITA would have recognized that there are only three apparent design options *to implement Chaudhri’s teachings* for using a drag gesture to move an application after shifting from the ‘normal mode’ to the ‘reconfiguration mode’ via a touch-and-hold gesture.”

Pet., 38.⁴ But the premise of Petitioner’s arguments that it is free to select “design options to implement Chaudhri’s teachings” assumes Chaudhri is silent and thus there is a “design choice” to fill in. But as discussed in Section VI.B.1, above, that premise is not correct because Chaudhri explicitly describes two non-continuous touches for the touch Petitioner maps to earlier claim elements and the touch Petitioner maps for meeting Limitations [1.l], [63.h], [124.g]. EX2005, ¶80-82.

Petitioner provides no reason beyond the inaccurate assertion that Chaudhri is silent for why a POSITA would even begin to explore modifications. Pet., 38-39, 62-67. Accordingly, both Petitioner’s Ground 1 and 2 modifications fail because Petitioner identifies no motivation to modify Chaudhri’s disclosure. EX2005, ¶85.

Petitioner’s Ground 1 and 2 mappings each also fail for further independent reasons. As to Ground 1, Petitioner’s motivations are generic “known and predictable” arguments that are not sufficient motivation. *See KSR Int’l Co.*, 550 U.S. at 418 (“[A] patent composed of several elements is not proved obvious merely

⁴ “First, the device and interface could be designed so that the user must keep their finger down ... Second, the device and interface could be designed so that the user must break contact ... Third, the device and interface could be designed so that the user may input the two gestures either as a single continuous touch or as consecutive touches.” Pet., 38-39.

by demonstrating that each of its elements was, independently, known in the prior art.”). Further, Petitioner’s assertion that “the most natural way to implement Chaudhri’s teachings would be to select the third” option “because the device and interface would [be] easier to use” (Pet., 39) inaccurately assumes Chaudhri’s silence and provides no analysis of Chaudhri’s actual implementation and why that is not the proper selection for Chaudhri.

Turning to Ground 2, according to Petitioner, “Os teaches a ‘continuous multipart gesture’ that includes a ‘tap and hold gesture’ combined with ‘subsequent lateral movement of the contact,’ all while the user’s finger remains down on the touch screen.” Pet., 66. In the terminology Petitioner coined in Ground 1, Petitioner’s characterization of Os matches the “first” option (“the user must keep their finger down”). *Id.*, 38-39. Given Petitioner already asserted that that a POSITA would select the “third” option (*id.*, 39), any argument that a POSITA would be motivated to select the “first” option rings hollow.

Further, a required continuous touch cannot be reconciled with Chaudhri itself. EX2005, ¶88. As discussed in Section VI.B.1, above, Chaudhri explicitly shows entering reconfiguration mode by touching one application icon, the Solitaire icon, and then performing a second touch and drag to move a second application icon, the Racing icon. Altering Chaudhri to require that the same continuous touch for both entering reconfiguration mode via touch of one icon and moving an icon

would remove the functionality shown across Figures 5A-5C that Chaudhri explicitly discloses. EX2005, ¶89. A POSITA would not be motivated to make such a modification. *Id.*

VII. LIMITATIONS [1.j-k], [63.f-g], [124.e-f]: A POSITA IMPLEMENTING MARTYN'S APP SNAPSHOT WOULD NOT ASSIGN A USER GESTURE AS IN THE PURPORTED COMBINATION

A. Petitioner's Convoluted Path To The Purported Combination

Petitioner's proposed combination begins with Chaudhri's device that already: (1) "transitions from the 'normal operation mode' to a 'reconfiguration mode' *when the user executes a touch-and-hold gesture*" on an application icon (Pet., 21), and (2) includes functionality, in reconfiguration mode, for "displaying a contextual .. sub-menu ... with options" and "presenting commonly used application functions" (*id.*, 18-19) *when the user taps a circle placed in the corner of the application icon* (EX1004, [0273]-[0274]). Ignoring these existing functions, and with the unapologetic goal "to successfully arrive at the '758 patent's claimed invention" (Pet., 16), Petitioner proposes a series of modifications inspired by nothing but hindsight.

First, Petitioner proposes "integrat[ing] Martyn's teachings about" (1) "providing a contextual shortcut sub-menu in the form of a 'summary window' common application functions (an 'App Snapshot')" and (2) calling the App Snapshot through one of a series of methods Martyn discloses: "a press and hold of

the select function *when an application name is highlighted.*” Pet., 16-17; EX1005, [0016]-[0017]. But, as discussed below, Petitioner identifies motivations only for the former, not the latter. Second, Petitioner does not propose actually using the selected gesture from Martyn, but rather using a different gesture that the Petition alleges in a single conclusory sentence to be “the analog to Martyn’s press-and-hold input on a touch screen.” Pet., 16.

Third, Petitioner proposes further modifications because “the POSITA would have sought a solution to the problem of distinguishing between the different gesture commands”—*the very problem created by the first modification.* *Id.*, 21. To resolve this conflict, Petitioner proposes assigning one gesture a long touch and another gesture a longer touch. *Id.*, 21-22. And, fourth, Petitioner proposes, not uncoincidentally to match the claim, calling the App Snapshot with a long touch, thereby *countermanding* Chaudhri’s disclosure of entering reconfiguration mode through a normal long touch on an application icon and, instead, requiring the user first call up the App Snapshot and then continue with a *longer* touch. *Id.* In support of those third and fourth modifications, Petitioner nominally points to Griffin, but does not identify any disclosure in Griffin of any motivation or advantages—just that Griffin happens to mention time thresholds. *Id.*

As discussed below, Petitioner’s series of modifications ignore the existing disclosures of Chaudhri that teach away from assigning a long touch to calling the

App Snapshot and already disclose a gesture for calling up what Petitioner describes as a context menu. Moreover, actually applying Petitioner's selected call up method from Martyn or applying numerous other alternatives Martyn discloses results in modifications not taught away from (and that do not meet the claims).

Accordingly, taking into account Chaudhri and Martyn's actual disclosures, a POSITA would be motivated to implement Martyn's App Snapshot using numerous call up methods, none of which match the claims. Petitioner's convoluted modifications, without which all Grounds fail are inferior to all of those methods and are justified only by hindsight.

B. Petitioner Fails to Identify A Motivation To Turn To Martyn For A User Gesture To Call The App Snapshot

Petitioner attempts a sleight of hand, relying on Martyn for *both* the App Snapshot and its disclosure of how to call the App Snapshot, but presenting motivations that address only the former. Pet., 15-19. Specifically, each of the four "motivations" Petitioner provides are limited to why a POSITA would include Martyn's App Snapshot. *Id.*, 16-19. Petitioner provides no reasons a POSITA would look to Martyn for its disclosure of how a user would call up the App Snapshot in Chaudhri, let alone why a POSITA would choose the specific option Petitioner selects over others Martyn discloses and are otherwise available. *Id.*

First: Petitioner asserts "it would have been axiomatic for a POSITA" to "*employ[] a contextual shortcut sub-menu* to augment a 'main menu listing some

or all applications’ of a mobile device” “in the context of Chaudhri’s user interface” because “sub-menus were exceedingly well-known solutions to [an] equally well-known problem.” *Id.*, 16-17. This does not purport to address why a POSITA would turn to Martyn’s disclosure of any particular user gesture. *Id.*; EX2005, ¶¶97-101.⁵

Second: Petitioner asserts that “*providing a sub-menu of options* for invoking common application functions, as Martyn teaches, would allow the user to accomplish recurring tasks more quickly and efficiently (e.g., with fewer user inputs).” Pet., 17. Again, this does not purport to address why a POSITA would turn to Martyn’s disclosure of any particular user gesture. *Id.*; EX2005, ¶¶102-105. To the contrary, the benefit of “accomplish[ing] recurrent tasks more quickly and

⁵ At his deposition, Dr. Cockburn argued that his discussion of this first motivation cross-referenced to a broader discussion of known context menus that he contended also discussed it being known. EX2007, 14:23-15:18. The Petition contains no such cross-reference, nor does the prose, which specifically states it is addressing a motivation for “providing a sub-menu,” state that it would be obvious to adopt any of Martyn’s methods of calling up an App Snapshot (let alone the one selected) because any method was well known. Pet., 16-17. Further, arguing an element is well known is not a sufficient motivation. *See KSR Int’l Co.*, 550 U.S. at 418.

efficiently” stems from selecting those tasks from the sub-menu, not from how the sub-menu is called. EX2005, ¶¶99, 103-104.

Third: Petitioner asserts that “Martyn identifies specific ‘advantages’ *of the ‘App Snapshot’*” and given “the similarities between Chaudhri and Martyn’s respective multi-functional mobile devices, the POSITA would have sought to predictably improve Chaudhri’s interface for accessing application functions in the same way Martyn did—i.e., *via a contextual shortcut sub-menu.*” Pet., 17-18. Again, this does not purport to address why a POSITA would turn to Martyn’s disclosure of any particular user gesture. *Id.*; EX2005, ¶¶106-107. All of Martyn’s description of benefits compare the improvement of using the App Snapshot to select a function without launch the application to accessing a function by first launching the application and then navigating to the function. EX1005, [0015]; EX2005, ¶108. As Petitioner asserts: “Where other techniques require the user to navigate between ‘many layers’ and/or only show ‘one option...at any one time,’ *Martyn’s App Snapshot* ‘brings together, in one summary window, a limited list of common functions,’ which ‘*yields many advantages in ease and speed of navigation.*’” Pet., 18 (quoting EX1005, [0004-0008], [0015]).

At deposition, Dr. Cockburn asserted Martyn’s passage “yields many advantages in ease and speed of navigation” means a POSITA “would absolutely understand that the mechanisms used to access that menu would also need to be easy

and speedy; otherwise, its objectives would not be met.” EX2007, 21:7-22. If Martyn is interpreted to advise a POSITA to minimize the steps to launch the App Snapshot as Dr. Cockburn asserted, that teaches away from adopting “a press and hold of the select function *when an application name is highlighted*” as Petitioner did. EX2005, ¶109. As discussed below, that method requires first navigating to highlight an application and then pressing and holding a select key, while other options disclosed by each of Chaudhri and Martyn that involve only a single step. *Id.*; EX1005, [0016]-[0017].

Fourth: Petitioner asserts a “POSITA would have been prompted to pursue a combination with Martyn based on Chaudhri’s suggestions to do so.” Pet., 18. Petitioner points to Chaudhri’s disclosure, in reconfiguration mode, of a “*notion of displaying a contextual* ‘pop-up dialogue’ *sub-menu* that ‘provides the user *with options* for modifying the selectable user interface object’” and asserts that suggestion “would have prompted a POSITA to seek out and apply the teachings from other references in the same field, like Martyn, that describe analogous implementations of contextual sub-menus to augment a user interface for a multi-functional mobile device.” Pet., 18-19 (citing EX1004, [0274], [0317]).⁶ Stated

⁶ Petitioner cites two paragraphs. Paragraph [0274], disclosing calling “object modification” “pop-up dialogue 5198,” which Petitioner asserts is a “contextual

otherwise, Petitioner asserts that because Chaudhri discloses a context menu (“objection modification” “pop-up dialogue 5198”) in one context, then it would be obvious to look for other context menus in other contexts and add them. *Id.* But, again, that motivation focuses on looking to Martyn for another type of context menu—an App Snapshot, not looking to Martyn for a method of calling that App Snapshot. EX2005, ¶111. Rather, as discussed in Section VII.B.2, below, the fact that Chaudhri already discloses a different type of context menu *with an associated user gesture* leaves a POSITA no reason to turn to Martyn for Petitioner’s desired calling method.

In sum, Petitioner fails to identify any motivation for why a POSITA would look to Martyn for its disclosure of how a user would call up the App Snapshot in Chaudhri, let alone why a POSITA would choose the specific option Petitioner selects over others Martyn discloses and are otherwise available. EX2005, ¶65. Absent any motivation, Petitioner’s combination, and thus all Grounds, fail because

‘pop-up dialogue’ sub-menu,” relates to the discussion above. Paragraph [0317] uses the words “contextual menu” in passing in describing something that could contain an input to be clicked on, but says nothing about how it was called. EX1004, [0317]; EX2005, ¶112; EX1007, 59:11-17 (“Paragraph 317 itself contains no explicit mention of how the contextual menu is -- is posted.”).

Petitioner has, at best, “demonstrate[ed] that each of its elements was, independently, known in the prior art,” which is insufficient to prove the Challenged Claims obvious. *See KSR Int’l Co.*, 550 U.S. at 418; *see also In re Rouffet*, 149 F.3d 1350, 1357 (Fed. Cir. 1998) (“Furthermore, rejecting patents solely by *finding prior art corollaries* for the claimed elements would permit an examiner to use the claimed invention itself as a blueprint for piecing together elements in the prior art to defeat the patentability of the claimed invention.”).

C. A POSITA Has No Reason To Turn to Martyn When Chaudhri’s Already Discloses A User Gesture For Calling a Context Menu

Petitioner contends that Chaudhri discloses “displaying a contextual .. sub-menu ... with options” in reconfiguration mode and “presenting commonly used application functions.” Pet., 18-19 (citing EX1004, [0247]); EX1003, ¶69. Specifically, Chaudhri discloses “in FIG. 5KKK each of the selectable user interface objects has an object modification target associated with it (e.g., action icon 5002-37 has a respective object modification target 5194).” EX1004, [0246]. A user calls the context menu, “object modification” “pop-up dialogue 5198,” through a “tap gesture 5196 at a location on the touch screen” corresponding to a circle placed in the corner of the application icon corresponding to “object modification target 5194.” EX1004, [0246]-[0247]; EX2005, ¶113; EX2007, 54:17-55:7 (“So the tap gesture is one example of the mechanisms used to activate the object modification target”).

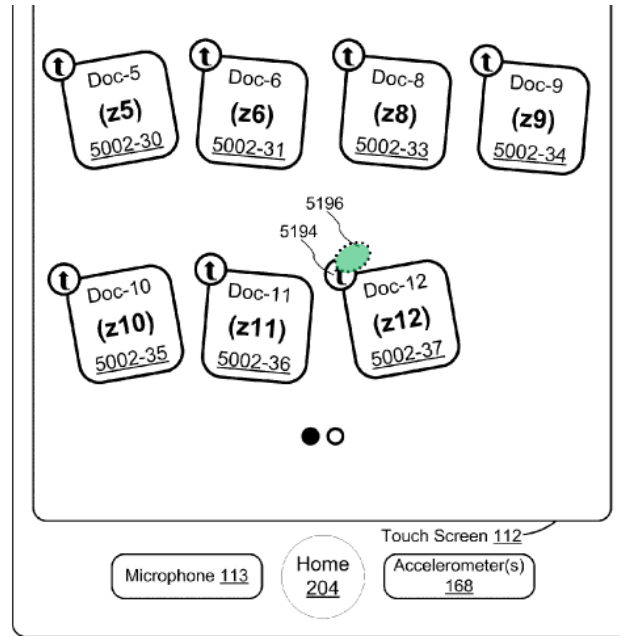


Figure 5KKK

EX1004, FIG. 5KKK (annotated); EX2005, ¶113.

A POSITA would understand this technique is readily applicable to calling an App Snapshot in normal mode. EX2005, ¶114. As annotated on Figure 5A, below, a POSITA would have understood that target regions (shown as red circles with “s” for “snapshot”) could be included in the upper lefthand corner could be included on icons associated with an App Snapshot when in normal mode just as the “object modification target” regions shown in Figure 5KKK. EX2005; ¶115; EX1004, [0273]-[0274]. Because the App Snapshot is called from normal mode, the “object modification target” regions shown in Figure 5KKK, in reconfiguration mode, would not cause any conflict. EX2005, ¶116. Then, just as in Figure 5KKK, the user could perform a “tap gesture” on the target region to call up the App Snapshot. EX2005, ¶117.

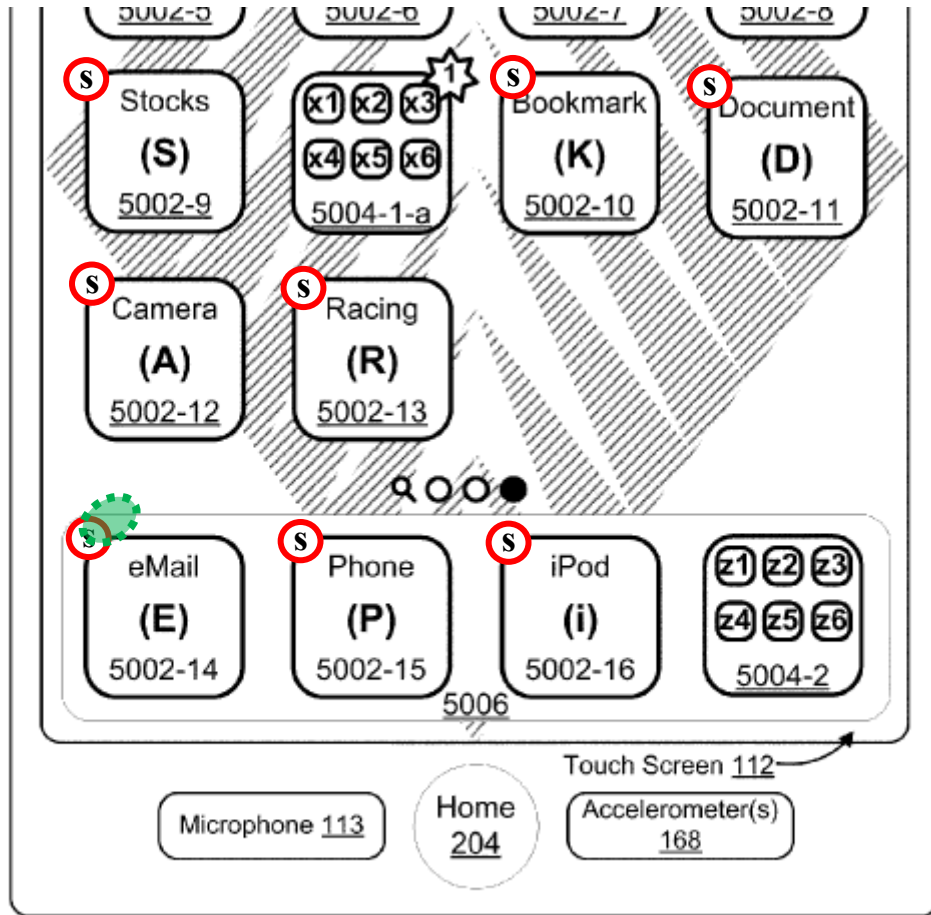


Figure 5A

EX1004, FIG. 5A (annotated); EX2005, ¶115.

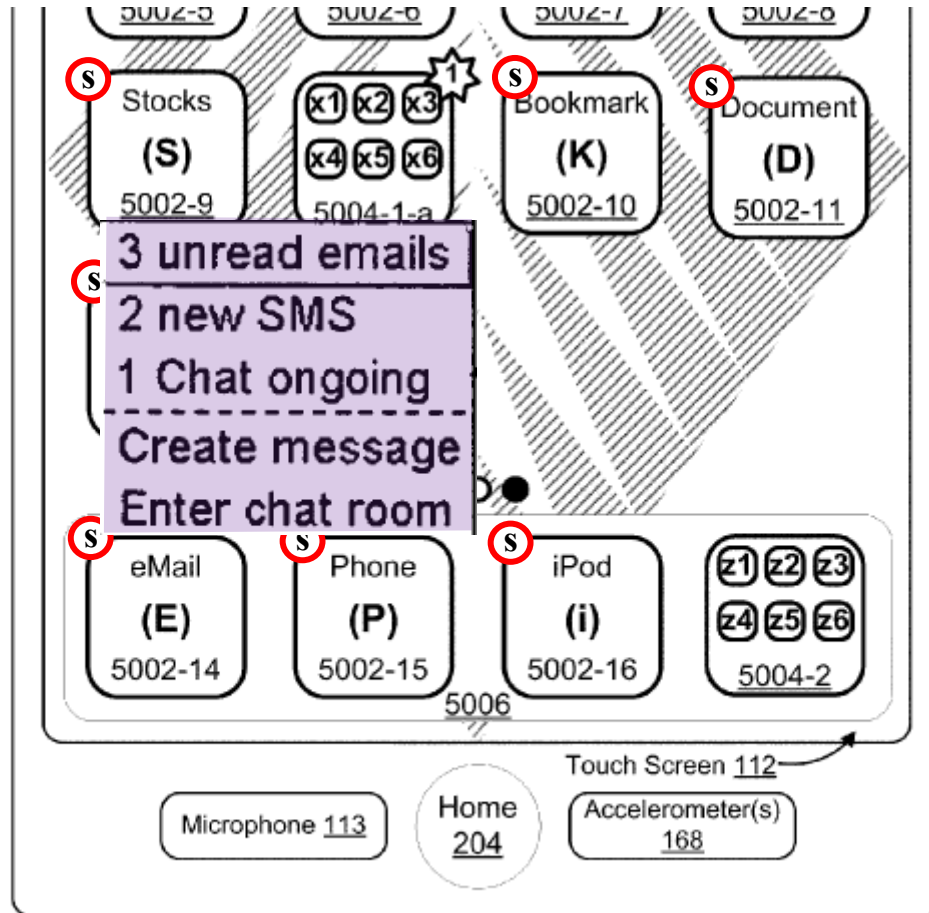


Figure 5A

EX1004, FIG. 5A (annotated after tap gesture); EX2005, ¶117.

Chaudhri's disclosure of a readily applicable method to call up a context menu teaches away from turning to Martyn's disclosure for that functionality because there is no need to do so. EX2005, ¶118. By failing to address this disclosure (despite citing to it for other purposes) or explain why it would not be sufficient, Petitioner fails to consider each reference as a whole. *In re Warsaw Orthopedic, Inc.*, 832 F.3d 1327, 1332 (Fed. Cir. 2016) (quoting *Panduit Corp. v. Dennison Mfg. Co.*, 810 F.2d 1561, 1568 (Fed. Cir. 1987)).

Further, the method Chaudhri already discloses is superior to Martyn's disclosure in multiple ways. Indeed, by adopting this specific method, the authors deemed it most compatible with Chaudhri's surrounding disclosure. Pet., 22; EX1003, ¶77; EX2005, ¶119. First, this method, as opposed to a long touch on an application icon as Petitioner proposes despite Chaudhri already assigning the long touch on an application icon to a different function, does not conflict with other user gestures. EX2005, ¶¶116, 120. Second, a tap on a target region is faster and simpler than the two-part (first navigate a highlight to an application and then long press a select function button) Petitioner proposes selecting from Martyn or the long touch Petitioner asserts is the "analog" to Martyn's actual disclosure. EX2005, ¶121. These are yet further reasons a POSITA would not be motivated to pursue the combination Petitioner proposes. EX2005, ¶122.

D. Chaudhri's Disclosure Assigning The Long Touch Gesture From A Different Function Teaches Away From Assigning That Gesture To Calling The App Snapshot

Petitioner readily admits that its proposal to assign a touch and hold gesture on an application icon to calling an App Snapshot—which Chaudhri had already assigned to a different function—creates a "*problem* of distinguishing between different gesture commands." Pet., 21. A POSITA would not be motivated to select an implementation that *creates a problem* when doing so does not solve any unsolved problem or have any benefit over other solutions that do not *create a*

problem. EX2005, ¶124-125. Rather, the fact that Chaudhri had already assigned the touch and hold gesture on an application icon to a different function **teaches away** from assigning that gesture to a second function. An implementation that **creates a problem** by taking a path from which the reference teaches away is not a suitable option. *Id.*; see *Meiresonne v. Google, Inc.*, 849 F.3d 1379, 1382 (Fed. Cir. 2017) (“A reference teaches away ‘when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken’ in the claim.”). Likewise, barreling forward with an implementation that conflicts with Chaudhri’s actual disclosures fails to consider Chaudhri as a whole. *In re Warsaw Orthopedic, Inc.*, 832 F.3d at 1332.

The facts here are readily distinguishable from *Intel Corp. v. Qualcomm Inc.*, 21 F.4th 784, 799 (Fed. Cir. 2021) and *KSR*. Unlike *Intel*, where the court held it was sufficient to adopt a “suitable option” even if it was not the “best option,” Petitioner’s proposal is **not a suitable option** because Chaudhri **teaches away** from pursuing such an option. See *Par Pharm., Inc. v. TWi Pharms., Inc.*, 773 F.3d 1186, 1197-1198 (Fed. Cir. 2014) (“Our precedent, however, does not require that the motivation be the best option, only that it be a *suitable* option **from which the prior art did not teach away.**”) (italics emphasis in original)). And, unlike *Intel*—where pursuing a goal (“to improve energy efficiency by turning off a portion of Der’s

circuit when not in use”) “triggered a known problem,” as discussed in Sections VII.B-VII.C—Petitioner identified no reason to adopt the touch and hold gesture for calling the App Snapshot, particularly where Chaudhri’s existing method for calling a context menu already provided a superior solution.

E. Numerous Available User Gestures That Are Consistent, Rather Than Conflict With Chaudhri and Martyn’s Teachings, Further Teach Away From Selecting a Gesture That Creates A “Conflict”

As discussed in Section VII.C, above, Chaudhri itself already discloses a user gesture—a tap on a target region in the top corner of an application icon—to call a context menu that does not conflict with Chaudhri’s other teachings. Moreover, as discussed below, applying the disclosures of Martyn and Chaudhri leads to numerous modifications that do not create a conflict. The availability of all of these options that do not create a conflict further teaches away from selecting a user gesture that creates a conflict. EX2005, ¶126.

1. Correctly adapting Martyn’s “press and hold of the select function when an application name is highlighted” does not match Petitioner’s modification and avoids conflict

As discussed in Sections V.B and VII.A, above, Martyn does not disclose calling the App Snapshot based on a touch and hold on an application icon. Rather, Petitioner points to Martyn’s disclosure of “a press and hold of the select function when an application name is highlighted.” Pet., 16-17; EX1005, [0016]-[0017]. Petitioner then, in a single conclusory sentence asserts the separate gesture is “the

analog to Martyn’s press-and-hold input on a touch screen like Chaudhri’s.” Pet., 16. The cited testimony of Dr. Cockburn is nothing more than a repeat of the conclusory claim in a footnote. EX1003, ¶63, n.3.

Contrary to Petitioner’s conclusory assertion, applying Martyn’s “press and hold of the select function when an application name is highlighted” to Chaudhri does not result in a touch and hold on an application icon. Martyn describes launching its Snapshot in a *two-step* process. *Supra* §V.B; EX1005, [0016]. First, a user navigates to a desired application using “[s]tandard highlight functions” before calling the App Snapshot. The method Petitioner relies upon explicitly states “a press and hold of the select function *when an application name is highlighted*,” indicating the required first step of navigating to the application. EX1005, [0016-0017]; EX2005, ¶128-129; EX2007, 39:3-12.

In applying Martyn’s disclosure to Chaudhri, a POSITA would first address the navigating step. EX2005, ¶130. Chaudhri discloses “[i]n some embodiments, device 100 may include a physical or virtual wheel (e.g. a click wheel) as input control device 116.” EX1004, [0093]. Chaudhri further disclose that “[a] user may navigate among and interact with one or more graphical objects (e.g. icons) displayed in touch screen 112 by rotating the click wheel or by moving a point of contact with the click wheel.” *Id.* Applying Martyn’s disclosure, a POSITA would implement the initial step of navigating the highlight using Chaudhri’s click wheel.

EX2005, ¶131. Next turning to the function key, Chaudhri further discloses using the click wheel for this purpose: “The click wheel may also be used to select one or more of the displayed icons. For example, the user may press down on at least a portion of the click wheel or an associated button.” EX1004, [0093]. Thus, to implement “a press and hold of the select function when an application name is highlighted” in Chaudhri, a user would use the click wheel to navigate a highlight to the desired application icon and then “press down on at least a portion of the click wheel or an associated button” in a long press to call up the App Snapshot. *Id.*, EX2005, ¶¶132-133.

Given Chaudhri specifically discloses navigation and select key functionality, a POSITA would use that functionality to implement Martyn’s disclosure. EX2005, ¶134. Petitioner’s conclusory claim that a different gesture is an “analog” ignores Chaudhri’s disclosure, specifically the availability of this functionality. *Id.* Further, this actual application of Martyn’s teaching constitutes another example of a method for calling an App Snapshot that does not create a conflict with Chaudhri’s existing disclosures. EX2005, ¶135.

2. *Martyn’s other methods of calling up the App Snapshot disclose further options avoiding conflict with Chaudhri*

The fact that applying other of Martyn’s disclosed call up methods also avoids a conflict further emphasizing the illogical and unnecessary nature of Petitioner’s insistence of adopting a call up method not found in the art and that conflicts with

Chaudhri's existing functionality. As an example, Martyn discloses beginning with the same "conventional navigation steps" discussed in Section VII.E.1, above, and then "call[ing] or fir[ing]" the Snapshot by ceasing scrolling such that "the highlight rest[s] on the name of an application in the App Launcher for a certain amount of time" until a "timeout" is reached. EX1005, [0016]-[0017]. A POSITA would apply this method to Chaudhri using the same "rotating the click wheel" to "navigate among and interact with one or more graphical objects (e.g. icons) displayed in touch screen" as discussed above. EX2005, ¶136-137; EX1004, [0093]. This method accurately follows Martyn's disclosure and does not result in a conflict.

As another example, a POSITA would have created multiple alternative non-conflicting call up methods using one of Martyn's "[o]ther selection processes": voice activation. EX1005, [0017]; EX2005, ¶138-139. Like Martyn, Chaudhri also discloses "accept[ing] verbal input for activation or deactivation of some functions through microphone 113." EX1004, [0085]. Thus, Martyn's disclosure of voice activation can be directly applied to Chaudhri. EX2005, ¶140. As Dr. Cockburn explained, there "would be more than one way to complete voice activation of [Martyn's] Snapshot," none of which would create a conflict with an already assigned input method. EX 2007, 34:2-6. "[I]n a two part process" a user could state "Messages. Show Snapshot," with a pause between stating "messages" and "show Snapshot." *Id.*, 35:6-9. Or "in a one-part process," a user could state "Show

message Snapshot” without a pause. *Id.* Each of these methods may be readily applied to Chaudhri to call up an App Snapshot without creating a conflict with Chaudhri’s existing functionality. EX2005, ¶141.

F. Petitioner Does Not Identify any Reason or Motivation to Turn to Griffin Beyond a Self-Induced Conflict

Petitioner argues that a “POSITA would have looked to incorporate Griffin’s teachings to evaluate a duration parameter of the user’s touch against certain predetermined criteria” because a “POSITA would have [needed to] [seek] a solution to the problem of distinguishing between [] different gesture commands” of “[1] a tap to launch the application, [2] a touch-and-hold to trigger a contextual sub-menu, and [3] another touch-and hold to transition into the reconfiguration mode.” Pet., 21. But Petitioner ignores that a POSITA only needed to seek Griffin, because Petitioner created a conflict by taking a specific calling method from Martyn. Pet., 16. Chaudhri already teaches what Griffin allegedly supplies—“evaluat[ing] a duration parameter of the user’s touch against certain predetermined criteria.” Pet., 16; EX1004, [0217] (“the device detects a contact ... for more than a predetermined period of time”), EX1004, [0153] (“receiving an input ... for more than a predetermined duration”); EX2005, ¶142-144. Absent creating a conflict there is no element for Griffin to supply. EX2005, ¶145. Therefore, Petitioner’s motivation is not based in the references or the knowledge of a POSITA but rather uses the claims as a map to reconstruct the invention.

VIII. PETITIONER FAILS TO MEET THE PLAIN LANGUAGE OF MULTIPLE DEPENDENT CLAIMS

A. Limitations [21] and [22]: Content Indication

Claim 21 recites “[t]he electronic device of claim 1, wherein: the first application is a content creation application; the one or more action option objects include a respective action option object that is associated with displaying at least a portion of previously created content; and wherein the one or more programs include instructions for: detecting selection of the respective action option object, and, in response to detecting selection of the respective action option object, *displaying the at least portion of previously created content.*” Claim 21 depends on claim 22 and recites “wherein the content creation application includes an email application, and *the displayed at a least portion of previously created content* includes a previous email message, for generating a reply thereto.

Petitioner maps “displaying the at least portion of previously created content” of Claim 21 to the text “3 unread emails” and “1 Chat ongoing” shown in Figure 3 of Martyn reproduced below. Pet., 43-44. Similarly, Petitioner maps “the displayed at least portion of previously created content” of Claim 22 to the same “unread emails” indication. Pet., 44.

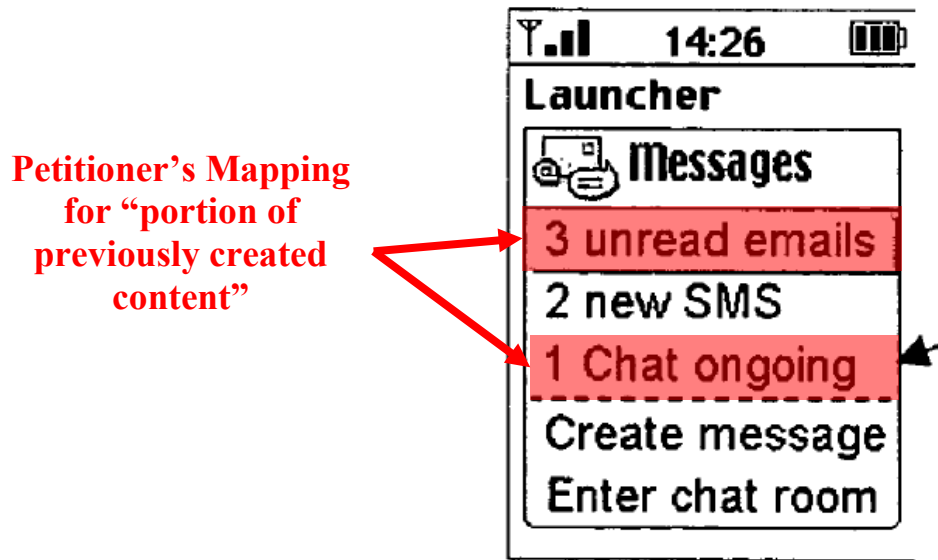


Figure 3

EX1004, Figure 3.

Neither “3 unread emails” nor “1 Chat ongoing” *display* any *portion of previously created content* as required by the claims. EX2005, ¶149-151. At best Martyn discloses indications informing a user that previously created content is available for viewing. As seen above, no content of the “unread emails” or “chats” are displayed. *Id.*

B. Limitation [35]: Multiple Moved Apps

Claim 35 recites “[t]he electronic device of claim 1, wherein the electronic device is configured such that, *in accordance with the determination that the first single-finger touch input meets the one or more movement criteria, multiple of the plurality of application icons is moved* in the foreground virtual display layer so that

the multiple of the plurality of application icons appear to float above the background virtual display layer.

Petitioner maps “multiple of the plurality of application icons is moved” “in accordance with the determination that the first single-finger touch input meets the one or more movement criteria” to Chaudhri’s disclosure that “the remaining *plurality of application icons* continue to jiggle (*move*) as the first application icon moves across the display” while in Chaudhri’s reconfiguration mode. Pet., 51.

Petitioner’s mapping ignores the claim language. The claim requires that multiple application icons are moved *when a movement criteria is satisfied*. The application icons that Petitioner attempts to map to are *already* jigging because Chaudhri’s device is in reconfiguration mode, which occurs *before* a detection of movement criteria (and which petitioner previously mapped to performing an operation when an *operation criteria is satisfied*, Pet., 34). EX2005, ¶154-155. Therefore, the application icons that Petitioner points to for this mapping are not moved when a movement criteria is satisfied. *Id.*

C. Limitation [47]: Interactive User Element

Claim 47 recites “[t]he electronic device of claim 1, wherein the electronic device is configured such that, during detection of the first single-finger touch input and prior to displaying the one or more action option objects, *a user interface element is displayed to appear at least partially behind the first application icon.*”

Petitioner maps “user interface element” to Chaudhri’s background wallpaper. Pet., 59. To support this mapping, Petitioner references a previous section describing Chaudhri’s background wallpaper as “providing the backdrop” for Chaudhri’s applications. Pet., 8; *see also* EX1004, [0268], [0366], [0368]. Petitioner also references its mapping for element [1.1], in this mapping Petitioner describes this background wallpaper as “static.” Pet., 37.

As discussed in Section IV.D, above, the plain and ordinary meaning of a “user interface element” is an element of a device that a user *interacts* with, which requires more than simply being a “*display* element.” EX2005, ¶¶55-58, 158. Petitioner does not identify any portion of Chaudhri disclosing that the user interacts with the background wallpaper, which Petitioner describes as merely “providing the backdrop.” Pet., 8, 59. Therefore, Petitioner’s mapping to a non-interactive element cannot meet the requirements of this Limitation.

D. Limitation [57]: Dynamically Generated Content

Claim 57 recites “[t]he electronic device of claim 1, wherein the electronic device is configured such that at least one of the one or more *action option objects includes content that is dynamically generated.*” Petitioner maps the one or more action objects that include content that is dynamically generated to the “common functions’ of the associated application[s]” in Martyn’s Snapshot. Pet., 59-60. Petitioner reasons that these “common functions” dynamically change because

Martyn's device learns over time which functions a user is interested in and accordingly varies what is shown to the user in Martyn's Snapshot. Pet., 60. But what Petitioner identifies is dynamic *selection* of existing available functions, not any content that is dynamically *generated*. EX2005, ¶160.

IX. CONCLUSION

For the foregoing reasons, the Petition should be denied in its entirety.

Dated: April 15, 2025

Respectfully submitted,

/Eagle H. Robinson/

Eagle H. Robinson

Reg. No. 61,361

Lead Counsel for Patent Owner

CERTIFICATE OF WORD COUNT

Pursuant to 37 C.F.R. § 42.24(a)(i), the undersigned certifies that this Paper—exclusive of the table of contents, table of authorities, certificate of service, and this certificate of word count—includes 10,735 words. The undersigned relies upon the word count generated by Microsoft Word.

/Eagle H. Robinson/

Eagle H. Robinson (Reg. No. 61,361)

CERTIFICATE OF SERVICE

Pursuant to 37 C.F.R. § 42.6(e), the undersigned certifies that on April 15, 2025 complete copies of this PATENT OWNER RESPONSE and EX2005-EX2008 were served on Petitioner via email (by consent), at the following email address(s):

IPR50095-0168IP1@fr.com

/Eagle H. Robinson/

Eagle H. Robinson, Reg. No. 61,361