

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.,

Petitioner

v.

SMITH INTERFACE TECHNOLOGIES, LLC,

Patent Owner

IPR2024-01256

Patent No. 11,740,727

DECLARATION OF RAJEEV SURATI, Ph.D.

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	PROFESSIONAL BACKGROUND	1
III.	COMPENSATION	10
IV.	LEGAL CONSIDERATION	10
A.	Obviousness	10
B.	Claim Interpretation	11
V.	Overview of Task & Basis for Opinions.....	12
VI.	Level of Skill in the Relevant Art	13
VII.	OVERVIEW OF THE '727 PATENT	14
A.	Independent Claim 1.....	16
VIII.	OVERVIEW OF THE PRIOR ART	17
A.	U.S. Patent App. No. 2008/0207188 ("Ahn").....	17
1.	Directional Key Operation	19
2.	Touch and Drag Operation	20
3.	Start Position Operation	22
B.	U.S. Patent App. No. 2007/0150842 ("Chaudhri-842")	23
1.	Unlocking a Device via Gestures.....	24
2.	Indication of Progress Towards Satisfaction of a User Input Condition.....	26
IX.	GROUND 1: DR. TERVEEN'S ANALYSIS FAILS TO SHOW OBVIOUSNESS OF CLAIM 1 BY AHN AND CHAUDHRI-842.....	28
A.	Limitation [1.d].....	28
1.	Dr. Terveen fails to Show Ahn's Graphical Shadow Moves.....	29
2.	Ahn-Chaudhri-842 would not increase magnitude of the change in appearance of Ahn's clock widget as a function of an increase in the distance magnitude of the gesture.....	31
3.	Ahn-Chaudhri-842 would not blur during detection of the gesture and before a completion of the gesture is detected	33
4.	Dr. Terveen's Proposed Ahn-Chaudhri Combination does not flow from the teachings of Ahn or Chaudhri-842	50
5.	Dr. Terveen's Proposed Modification Would Not Achieve the Functionality Allegedly Motivating the Proposed Combination	60

X.	CONCLUSION	70
----	------------------	----

I. INTRODUCTION

1. My name is Dr. Rajeev Surati, Ph.D. I have been retained by Smith Interfaces Technologies, LLC as an independent expert consultant in this inter partes review (“IPR”) proceeding before the United States Patent and Trademark Office (“PTO”).

II. PROFESSIONAL BACKGROUND

2. All of my opinions stated in this Declaration are based on my own personal knowledge and professional judgment. In forming my opinions, I have relied on my knowledge and experience in designing, developing, and researching the technology referenced in this Declaration.

3. I am over 18 years of age and, if I am called upon to do so, I would be competent to testify as to the matters set forth herein. I understand that a copy of my current curriculum vitae, which details my education and professional and academic experience, is being submitted as EX2002. The following provides a brief overview of some of my experience that is relevant to the matters set forth in this Declaration.

4. I hold a Doctor of Philosophy (Ph.D.) degree in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology (“MIT”) (awarded in 1999) with a Grade Point Average of 5.0/5.0. I obtained a Master of Science degree in Electrical Engineering and Computer Science from MIT (awarded

in 1995) with a Grade Point Average of 5.0/5.0. I have a Bachelor of Science degree in Electrical Engineering from MIT (awarded 1992), graduating with a Grade Point Average of 4.9/5.0.

5. My Ph.D. thesis was entitled “Scalable Self-Calibrating Technology for Large Scale Displays.” My master’s thesis was entitled “Practical Partial Evaluation.” My undergraduate thesis, for which I received MIT EECS’s William A. Martin thesis prize for best undergraduate thesis, was entitled “A Parallelizing Compiler based on Partial Evaluation.” I was also awarded the Department of Energy’s highly-selective Computational Science Fellowship in 1995, which funded my Ph.D. studies.

6. In 1989, I began programming embedded systems and, in 1995, built one of the first publicly available Internet Coke Machines. During that time, I worked on various user interfaces. For example, one to augment the coke machine to dispense coke and another that became one of the early ecommerce mechanisms, using standards-based protocols for tracking user consumption, payments, and drink stock.

7. In 1997, while at MIT pursuing my Ph.D., I started a company called Flash Communications. At Flash, I invented an instant messaging platform focused on enterprise needs. I had the idea while observing the popularity of the Zephyr Instant Messaging Service at MIT beginning in 1988. Attracted by Flash’s

enterprise focus, Microsoft acquired Flash in 1998. With Microsoft, we built both a client and server product. The basic protocol we invented became the foundation for the well-known XMPP protocol that was widely used in the mid-2000s among instant messaging providers. I developed both the client and server products and dealt with many, if not all, of the issues one might face when implementing user interfaces for instant messaging systems. In fact, we helped to build Microsoft's famous MSN product, which adopted various user interface elements we created.

8. As a result of this experience, I became heavily involved with User Interface design on desktop machines, experience that I have continued to develop by building desktop, mobile, and web-based applications used by millions of people.

9. After graduating, I joined Microsoft (as was required by Microsoft in its acquisition of Flash Communications), where I worked on both client and server technologies related to instant messaging. My work spanned both the Microsoft Exchange Instant Messaging product that was released in 2000 and MSN Messenger.

10. From 2000 to 2007, I cofounded, led, and sold a business called photo.net to Namemedia, which is now part of GoDaddy.com. In early 2000, Photo.net was considered a top 1,000 website and it received large amounts of traffic. I built the site from its beginning as a single computer installed in a datacenter to a full rack of computers in that same data center. As a result, I became very familiar with the careful design and programming one must employ to build

and maintain systems that enable both engaging user interfaces and experiences at scale.

11. At photo.net, we prototyped many fundamental Internet community features, such as photo sharing, social networking, and memberships. The photo.net system was written on top of Oracle SQL Database and served up many millions of web pages each day (corresponding to millions of records) with high-volume inserts. The system also incorporated an ecommerce platform used to bill subscriptions, track users, etc. I spent seven years running the site, both writing new features driven by the Oracle SQL database and maintaining tables containing the many millions of records generated by the site. I also served as the database administrator, which provided me with extensive operational experience running such a system. I dealt with everything from database query speedup to the more mundane day-to-day issues, such as the maintenance of relational databases, including backups. In addition to the web interface, we also ran a mobile version of the product based on WAP, which I helped to develop.

12. After developing one of the first photo sharing sites, we further increased user engagement by designing an innovative photo rating interface. Our photo rating system became a huge draw for the site. The rating system increased engagement by four orders of magnitude, ultimately resulting in Sergey Brin, cofounder of Google, asking photo.net to provide imagery to Google for a screen

saver application. Joichi Ito, former head of the MIT Media Lab and a famous internet luminary called it “the best photo sharing site” in 2001.

13. I also innovated user interface technology at Nexaweb, where I was an investor and technical advisor. Nexaweb designed User Interfaces that worked well over the web. At Nexaweb, I spent a great deal of time solving underlying issues related to implementing a UI framework both in Java and Javascript. We also innovated a mechanism using server push prior to the AJAX becoming widely adopted.

14. In 2004, I started Scalable Display Technologies based on the technology in my Ph.D. thesis and the related patent (U.S. Patent No. 6,456,339) entitled “Super-resolution Display.” As part of my Ph.D. thesis, I designed a special display that allows one to create ultra-high-resolution displays composed of multiple projectors tiled with a slight overlap. The system uses camera-based feedback to create an inverse map that drives the system such that a person can only see a continuous, seamless display. There is no visible bezel or overlap. During this project, I became familiar with how and why to drive content onto the display. Thus, I also became familiar with content encoding/decoding, user interfaces for driving large displays, multimedia content storage, high resolution imagery, networks, recording, GPUs, storage of content, etc.

15. At Scalable Display Technologies, I developed the core algorithms and implemented—with a team—the entire software system, which included the core user interface, machine vision and embedded components, and a multimedia playback system. Our first customer was Pixar. Today, the software is primarily used by the US military and commercial theme parks, such as Disney and Universal. Scalable is well-known in the industry for having some of the easiest and intuitive software to use.

16. In parallel with my work at Scalable, I lectured at MIT on many subjects, including the Android operating system. In 2008, I worked with a group of MIT students on the development of mobile applications for Android with Rich Miner, a cofounder of Android. Since many students created applications involving Google Maps and localization, I became familiar with these technologies. I also served as a lecturer and mentor at MIT Play Labs, which was an incubator for augmented reality (AR) and virtual reality (VR) software for mobile handsets and headset applications. Notably, in addition to mobile handsets, Android runs on the Oculus headset. As part of that program, I worked with several startup companies on mobile applications, including a startup that developed applications for Telegram, which is a cross-platform, cloud-based instant messaging system.

17. Since 2014, I have worked as an independent consultant for several companies including NEC, Hitachi, Hi Marley, and Estee Lauder.

18. In 2018, I became a senior partner at nCent Labs. In that role, I consulted on the development of an incentive-market-based platform for block chains and cryptocurrency. Another part of my work at nCent Labs focused on the development of SMS messaging applications for the nCent platform. In that part, I developed a full stack application around the protocol.

19. From 2019 to 2020, I served as a Technical Lead of the Skunkworks at Hydrow, which is a startup company that develops indoor rowing machines. In this role, I worked on special projects, including the development of a VR experience using Magic Leap and Oculus that immersed users in a team-based rowing crew on a scull using VR cinematography. I leveraged my experience with user interfaces and Human-Computer Interaction (“HCI”) to develop a system that gives people a sense of rowing on the water with others, while in reality, they were rowing on a rowing machine. We accomplished our goal through a clever use of cinematography. The result was quite compelling and increased user engagement with their exercise program.

20. In 2020, I started a company called Skyline Nav AI Inc. that develops technology that uses a visual location (i.e., the skyline) to geo-locate the place a picture was taken. At Skyline, I built and now maintain a full user experience centered around a workflow for people to snap a picture and use the image’s skyline and downloaded 3D mode to localize a person. I have written applications for both

Android and iOS that communicate with a backend server running the machine vision algorithms.

21. Over the past decade, I have served as a technical consultant and expert witness on patent litigation matters. In the course of this work, I have provided consulting services to a wide variety of technology companies, including BritishTelecom, Apple, IBM, Philips, Shopify, Zillow, Polaris Powered Technologies, Amazon, Salesforce, Hitachi, Slack, Harris Teeter, and others.

22. During the course of my career, I was the inventor of the subject matter for approximately 15 U.S. Patents. I have also received additional patents, including: U.S. Patent No. 8,817,111, entitled “System and method of calibrating a display system free of variation in system input resolution,” U.S. Patent No. 8,994,757, “System and method for providing improved display quality by display adjustment and image processing using optical feedback,” U.S. Patent No. 9,215,455, “System and method of calibrating a display system free of variation in system input resolution,” U.S. Patent No. 9,369,683, “System and method for calibrating a display system using manual and semi-manual techniques,” and U.S. Patent No. 9,497,447, “System and method for color and intensity calibrating of a display system for practical usage.”

23. I am also a co-inventor of patented technology related to instant messaging, including U.S. Patent No. 5,943,478, which describes associated

technology that I developed related to pop-up, two-way messaging over the Internet. While at Microsoft, I was an inventor on several patents, including: U.S. Patent No. 6,260,148 relating to methods and systems for message forwarding and property notifications using electronic subscriptions; and U.S. Patent Nos. 6,415,318 and 6,604,133 relating to inter-enterprise messaging systems using bridgehead servers. Aspects of these patents relate specifically to messaging and notification technology in telecommunications systems.

24. I serve on the advisory boards of several technology companies, including Paneve, which develops general purpose ASICs coupled with compiler technology; Nexaweb, which develops real-time web application frameworks using HTTPS; Antix Labs, which develops compiler technology for a universal gaming platform; and Permabit, which develops content-addressable storage.

25. In addition to my above experience, I have published papers on subjects relating to computing systems, computer network communications, databases, and other subjects within the realm of electrical and computer engineering. I have also received several awards for my contributions as an inventor and entrepreneur, including the Global Indus. Technovator Award 2009 and Laureate of 2009 Computer World Honors Program. My publication and award history are listed in more detail on my CV.

III. COMPENSATION

26. My work on this matter is being billed at my customary rate of \$700 an hour. Also, I am being reimbursed for reasonable expenses I incur in relation to my services. I have no pecuniary interest in the outcome of this proceeding. I understand I will be paid regardless of the outcome of any proceeding in which my work is used.

IV. LEGAL CONSIDERATION

27. I am not an attorney. My understanding of the law is based on information provided by counsel for Smith Interface Technologies, LLC.

A. Obviousness

28. I understand that a claimed invention is obvious and, therefore, not patentable if the subject matter claimed would have been obvious to a person of ordinary skill in the art at the time that the invention of the '727 Patent (a "POSITA"), which I have been asked to treat as August 5, 2011.

29. I understand that a claim can be obvious in view of a single prior art reference (e.g., via modification of that prior art reference) or multiple prior art references (e.g., via a combination of two or more prior art references), if such a modification or combination was within the skill of a POSITA. I understand that there must be some articulated reasoning with some rational underpinning to support a conclusion of obviousness. I also understand that to establish a finding of obviousness one must show that one of ordinary skill in the art would have had a

motivation to combine the prior art references to produce the claimed invention and a reasonable expectation that the combination would be successful.

30. I further understand that exemplary rationales that may support a conclusion of obviousness include: (1) simply arranging old elements in a way in which each element performs the same function it was known to perform, and the arrangement yields expected results, (2) merely substituting one element for another known element in the field, if the substitution yields no more than a predictable result, (3) combining elements in a way that was “obvious to try” because of a design need or market pressure, where there was a finite number of identified, predictable solutions, (4) that design incentives or other market forces in a field would have prompted variations in a work that were predictable to a person of ordinary skill in the art, and (5) that there was some teaching, suggestion, or motivation in the prior art that would have led a POSITA to modify or combine prior art references to arrive at the claimed invention.

B. Claim Interpretation

31. I understand that a claim term is interpreted according to its ordinary and customary meaning as a POSITA would have understood the term in light of the surrounding claim language, other claims, the specification, and the patent’s prosecution history, which are referred to as intrinsic evidence. I also understand that prior art references cited in the patent’s prosecution history are considered

intrinsic evidence. I further understand that evidence outside the patent and its prosecution history (e.g., dictionaries and technical articles), may inform the context in which a POSITA would have understood the claims of a patent. I understand this ordinary and customary meaning applies absent unique circumstances, such as where a patent clearly expresses an intent to set forth a special meaning for a term.

V. Overview of Task & Basis for Opinions

32. I have been asked to review the '727 Patent. I have been asked to provide opinions related to certain issues from the perspective of a person of ordinary skill, having knowledge of the relevant art, as of August 5, 2011, and—except where otherwise noted—the opinions stated in this declaration are from that perspective. The qualifications and abilities of such a person are described in paragraph 38 below.

33. More particularly, I have been informed that this proceeding is currently at a preliminary stage during which the Patent Trial and Appeal Board (the Board) will decide whether the Petition establishes a reasonable likelihood of at least one claim of the '727 Patent being invalid. I also understand that the Board may review the Petition under a stricter “compelling merits” standard that requires the Petition to make a stronger showing of invalidity; however, my analysis is not performed using this higher standard. At this preliminary stage, I have been asked to focus my analysis on the Petition’s proposed invalidity theories for the '727 Patent’s independent claim 1.

34. My opinions are based on my education, training, and experience as well as items that I reviewed to prepare my opinions, including the Petition, the documents listed in the Exhibit List included with the Petition, and any other items that I reference in my below analysis.

35. I understand that other issues may arise that require further explanation, and I will provide that explanation if appropriate. As a result, I respectfully reserve the right to update and supplement this declaration and the information and opinions provided herein.

VI. Level of Skill in the Relevant Art

36. I understand that a patent must be analyzed from the perspective of a person of ordinary skill in the art (“POSITA”) as of the time of invention of the patent, which is typically considered to be the patent’s earliest filing date. While I do not necessarily agree with every aspect of Dr. Terveen’s explanation of the level of ordinary skill in the art, I do not currently believe the aspects of that explanation I would change or clarify would impact the analysis set forth in this declaration. For purposes of this declaration, my opinions are therefore provided from the perspective of the POSITA described by Dr. Terveen’s declaration (EX1003) as of August 5, 2011, which is produced below:

“a Bachelor of Science degree in computer science, computer engineering, electrical engineering, human-computer interaction, or a

comparable field and at least two years of professional experience in the design and development of graphical user interfaces (GUIs), human-computer interfaces, or equivalents thereof. Such experience could be obtained through research and study in a graduate program or through comparable exposure through industry employment, and additional years of experience could substitute for the advanced-level degree.”

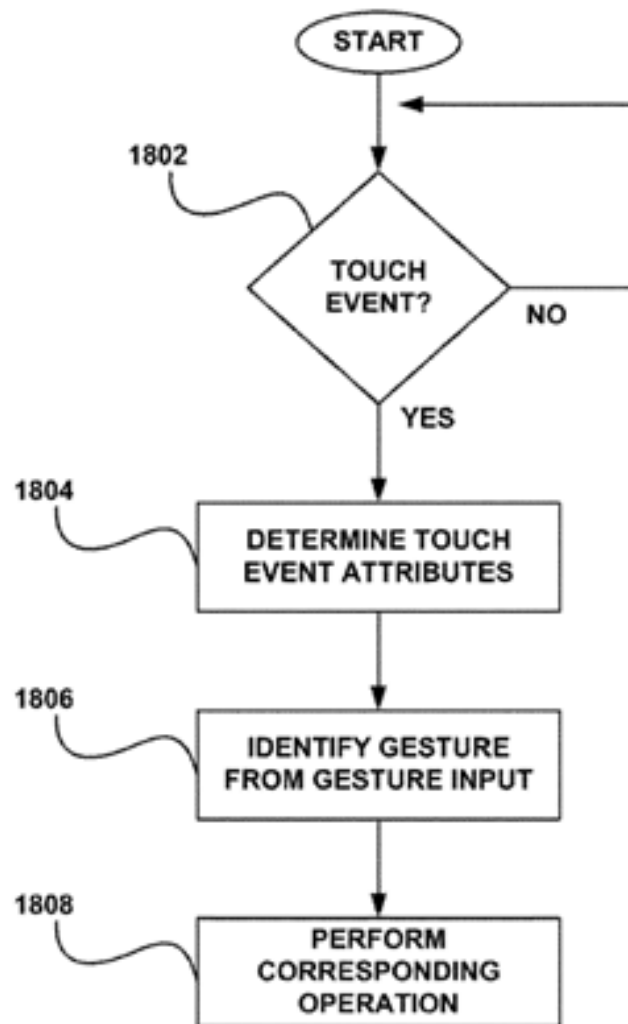
EX1003, ¶28.

VII. OVERVIEW OF THE ’727 PATENT

37. The ’727 Patent describes devices, methods, and graphical user interfaces for manipulating user interface objects with visual and/or haptic feedback. *See* EX1001, at Title.

38. The ’727 Patent describes a mobile device having various components, including user interface devices such as a touch screen. EX1001, 13:41-58. In operation, signals can be received in association with the touch interface (e.g., touch screen) and can correspond to a signal indicative of a touch interface being touched. *Id.*, 14:11-33. The user experience can then be altered utilizing these touch signals. *Id.*, 15:19-31, FIG. 3.

39. An example of these operations can be seen in FIG. 18, where attributes of touch events are detected, and gesture inputs related to these touch event attributes (both initial and over time) can be identified. EX1001, 56:8-35.



EX1001, FIG. 18.

40. These attributes can include the duration and/or distance of a gesture because often different gestures will be indistinguishable based on initial contact. EX1001, 56:8-35, 60:43-47. Once the gesture is identified, a corresponding operation can be performed (step 1808). EX1001, 56:50-51. Such

operations can include the modification of one or more display attributes such as blurring the display attributes. EX1001, 57:9-10, 68:37-52.

41. In some operations, “a user may be given feedback as a gesture is performed.” *Id.*, 59:23-24. For example, some “gestures may be associated with operations that are carried out as functions of the gesture input” EX1001, 58:58-59:3. In this way, the device can provide a user with feedback as a gesture is performed. *See* EX1001, Title (visual feedback), 59:23-29, FIG. 18 (step 1802 showing continuous detection of touch event as gesture is performed).

A. Independent Claim 1

42. Independent claim 1 is directed to a “device” having one or more processors that are configured to execute instructions that cause the device to perform functions. EX1001, 102:31-50. Specifically, claim 1 recites

[1.a] A device, comprising: at least one non-transitory memory; a touch screen; and one or more processors in communication with the at least one non-transitory memory, and the touch screen, wherein the one or more processors execute instructions in the at least one non-transitory memory, to cause the device to:

[1.b] display an object and at least one other object;

[1.c] detect at least part of a gesture on the touch screen; and

[1.d.i] during detection of at least a portion of the gesture before a completion thereof is detected, blur, based on a change in a distance

magnitude of the gesture being detected on the touch screen to thereby move the object on the touch screen, at least a portion of the at least one other object that is not overlapped

[1.d.ii] such that a magnitude of the blur itself, and not a magnitude of an area of the touch screen that is blurred, is increased as a function of an increase in the distance magnitude.

EX1001, 102:31-50.

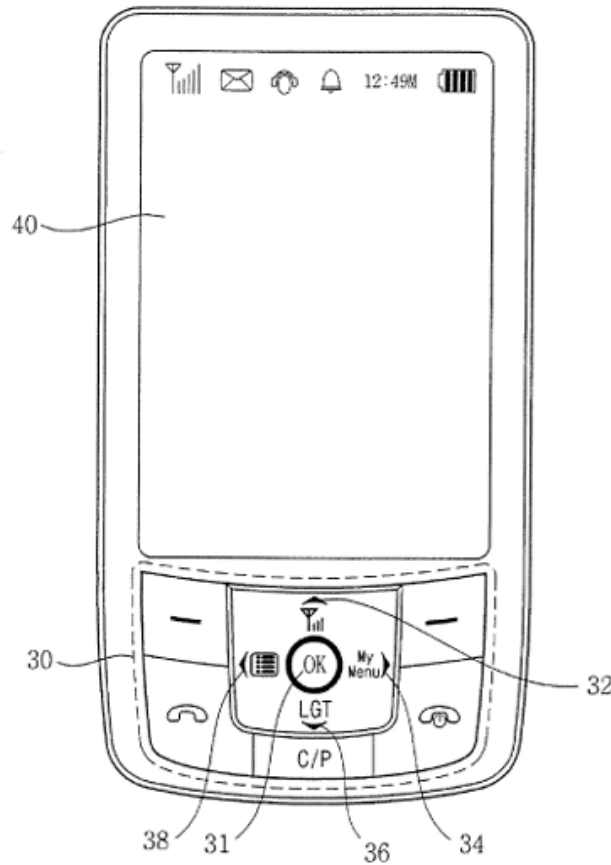
VIII. OVERVIEW OF THE PRIOR ART

43. Dr. Terveen primarily relies on two cited references for his invalidity analysis. See EX1003, ¶62 (the Ahn-Chaudhri combination).

A. U.S. Patent App. No. 2008/0207188 (“Ahn”)

44. As depicted in FIG. 2 (inset below), Ahn discloses a mobile communication terminal that includes a display 40 along with a key input device 30 and a touch input device (e.g., element 10 in FIG. 1). EX1004, Abstract, ¶[0027].

FIG. 2



45. Ahn's proposed operation attempts to solve the problem where device characteristics are unavailable when a menu list is displayed. EX1004, ¶¶[0005]-[0006]. In conventional browsers, the standby screen disappears when a menu screen is displayed. Thus, Ahn proposes to provide menus that only take up a portion of the display so that the standby screen can still be displayed concurrently with the menu.

46. Against this backdrop, Ahn discloses a method of displaying menus in a mobile communication terminal, where each menu can correspond with a selection of one of a plurality of specific directions associated with the menus. EX1004,

Abstract (“menu corresponding to the selected specific direction on the standby screen”). In Ahn, the menu is displayed after an input (e.g., after selection of direction key (¶[0034]) or after a touch and drag operation (¶[0102])) is completed. If the input is not completed, the menu will not open and will remain closed (i.e., not displayed). See e.g., EX1004, ¶[0116] (menu can be displayed “if the touch and drag operation of the user *is completed*”).¹

47. As described in more detail below, Ahn discloses three separate methods for displaying its menu via the use of directional keys, a simple touch and drag operation, and via a start position operation.

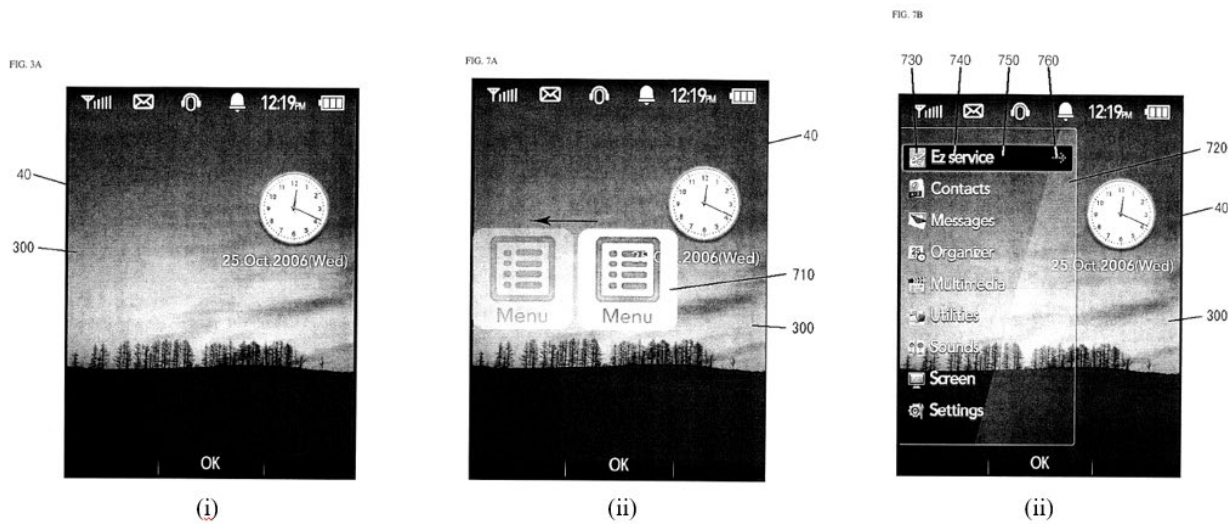
1. Directional Key Operation

48. In a first embodiment, Ahn describes a method of displaying a menu via a selection of a specific direction on a direction key. See EX1004, ¶¶[0030], [0034]-[0035]. As shown in FIGs. 7A-7C, a menu (e.g., setting menu 720/820) can be accessed “when a leftward direction key is selected.” *Id.*, ¶[0062]. This operation starts “when the standby screen 300 is displayed on the display 40.” *Id.*, ¶[0063]. After the “user selects the leftward direction key 38,” an icon 710 is “displayed around the center of the standby screen 300” and then “disappears while moving to

¹ All emphasis added unless otherwise noted.

[the] left” side of the screen. *Id.* A menu associated with the direction of the user’s completed input is then displayed. *Id.*

49. This operation is illustrated in greater detail below, in which three graphical settings are displayed to the user.



First, the standby screen 300 is displayed. EX1004, ¶[0035]. Next, after the left direction key is pressed (i.e., the gesture is completed), an icon 710 is displayed and fades to the lefthand side of the screen to provide the user with feedback as to the operation associated with the left direction key. *Id.*, ¶[0063]. Finally, once the icon 710 disappears, a menu 720 is displayed on the lefthand side of the standby screen 300. *Id.*

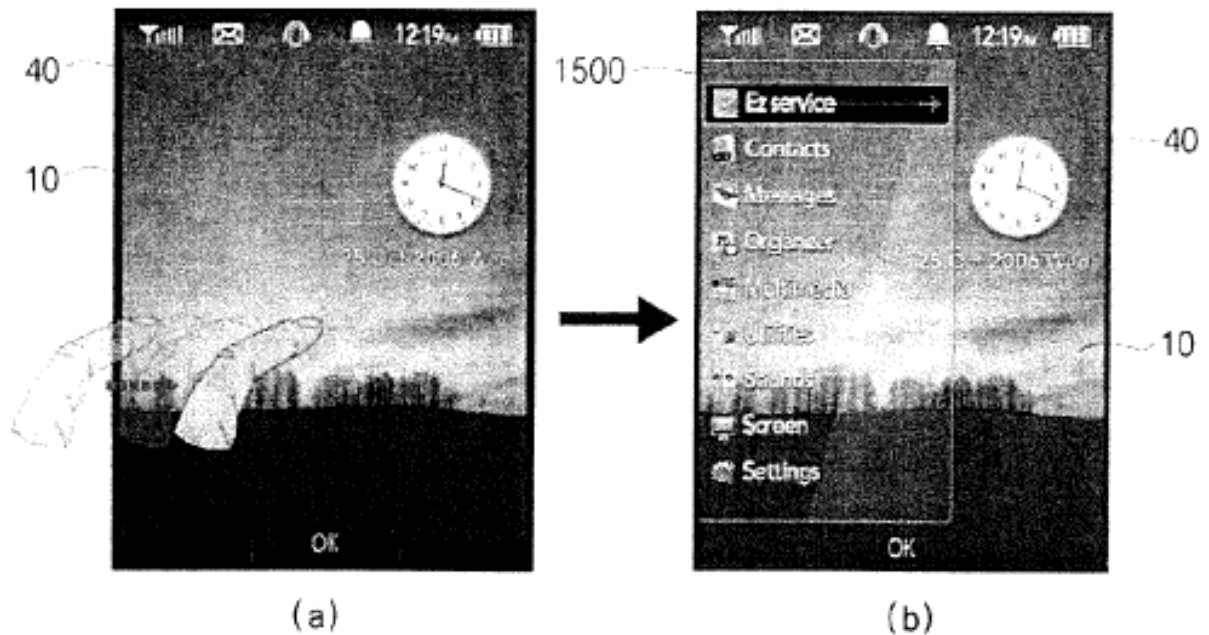
2. Touch and Drag Operation

50. In another embodiment, Ahn describes a method of displaying a menu via “a touch and drag operation of a specific direction.” *Id.*, ¶[0101], see also ¶¶[0102]-[0104]. As shown in FIGs. 15A-15B of Ahn (inset below), “if the user

IPR2024-01256 20 - Patent No. 11,740,727

touches screen 10 and drags the touch in a rightward direction, a menu list 1500 matched to a right direction is displayed in the display 40.” *Id.*, ¶[0103].

FIG. 15



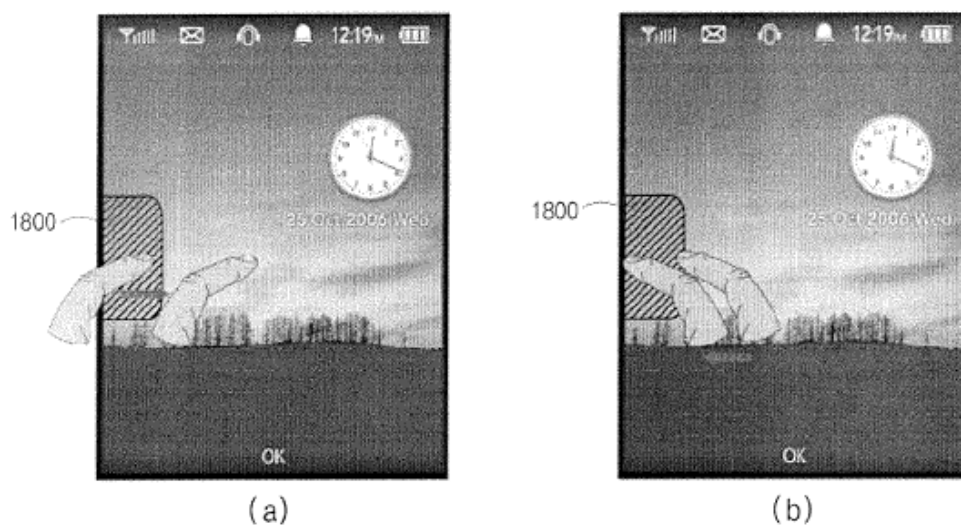
51. In this operation, “[a] controller 50 detects a touch and drag operation of a specific direction in the touch screen 10.” *Id.*, ¶[0107]. Then “controller 50 determines whether a drag distance of the detected touch and drag operation is a first distance or more.” *Id.*, ¶[0109]. *After* the user performs the touch and drag gesture and the controller compares the drag distance to a threshold, the menu can then be displayed. *Id.*; *see also* FIG. 16 (detecting touch at step 1610 and displaying menu at step 1630). “That is, if the touch and drag operation of the user *is completed*, the controller 50 determines a distance of the touch and drag operation, whereby a

specific menu list corresponding to the distance can be displayed in the display 40.”

EX1004, ¶[0116].

3. *Start Position Operation*

52. In another embodiment, Ahn describes methods of “notifying a start position of a touch and drag operation to a user,” as illustrated in FIGs. 18a and 18b (inset below). *Id.*, ¶[0118].



53. In these embodiments, a “graphical shadow 1800,” instead of a menu, is displayed based on a user’s touch and drag operation. Unlike the directional key and touch and drag operations described above, this operation does not open a menu that corresponds to the direction of a user’s input and, instead, menu 1500 is opened based on a left and right swipe. EX1004, ¶¶[0118]-[0119] (opening menu list 1500 when “the user drags in a right direction after touching” and when “the user touch drags to the left”).

54. In a first operation, “if the user drags in a right direction after touching the screen, a graphical shadow 1800 is displayed in the display 40.” *Id.*, ¶[0118]. Then, after the drag in the right direction is completed, the user perform a drag action on the graphical shadow 1800 to display menu list 1500, as illustrated in FIG. 18A. *Id.*

55. In a second operation, the menu list 1500 can also be displayed “if the user touch drags to the left.” *Id.*, ¶[0119]. In this operation, the user touch drags to the left to display the graphical shadow 1800 and, after the gesture is completed, the user may then press the graphic 1800 with a finger. *Id.* After the user selects the graphical shadow 1800 (when activated with a left swipe), the graphical shadow “disappears in the display 40, and the menu list 1500 can be displayed.” Thus, Ahn’s Start Position operation provides an alternative manner of handling the detected touch and drag gestures than the Touch and Drag Operation described in VIII.A.2, above.

B. U.S. Patent App. No. 2007/0150842 (“Chaudhri-842”)

56. Chaudhri-842 discloses “[a] device with a touch sensitive display [that] may be unlocked via gestures performed on the touch-sensitive display.” EX1005, Abstract. The primary focus of Chaudhri-842 is a method for “unlocking [a] device” via a “predefined gesture” where the method includes “moving [an] unlock image to a predefined location and/or moving the unlock image along a predefined path.”

EX1005, Abstract. Chaudhri-842 describes three different operations, each broken up into its own section: (1) Unlocking a Device via Gestures (EX1005, ¶¶[0052]-[0079]); (2) Indication of Progress Towards Satisfaction of a User Input Condition² (EX1005, ¶¶[0080]-[0098]); (3) User Interface Active States Corresponding to Events or Applications (EX1005, ¶¶[0099]-[0108]).

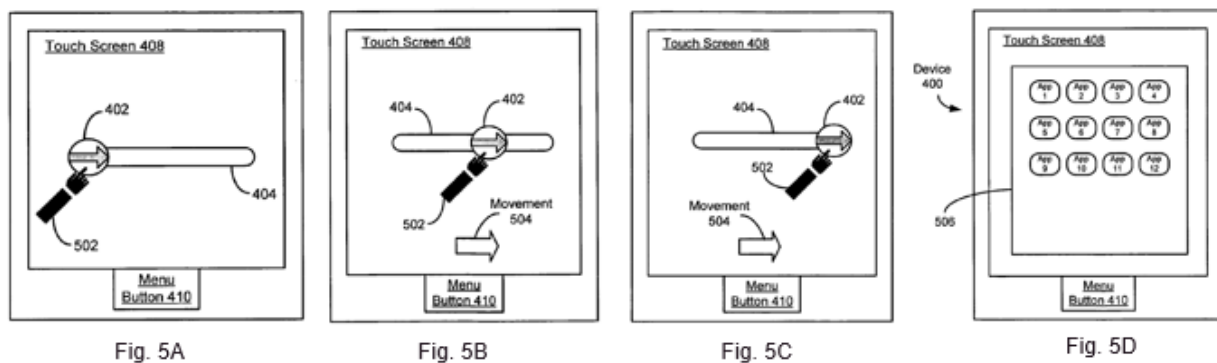
1. Unlocking a Device via Gestures

57. In this operation, Chaudhri-842 discloses a process for unlocking a device via transitioning from a lock screen to a separate interface. For this embodiment, Chaudhri-842 provides two examples of this transition: (1) between a lock state and an unlock state (EX1005, ¶¶[0048], [0052]-[0078]) and (2) between a first and second application (EX1005, ¶¶[0048], [0079]).

58. In a first operation, Chaudhri-842 describes a method of unlocking a device using a predefined gesture via transitioning from a lock screen to a separate interface. *See* EX1005, ¶¶[0074]-[0076]. FIGs. 5A-5D (inset below) show a first scenario in which a user transitions from a lock state to an unlock state. This operation starts when a user “touch[es] a touch screen 408 of device 400 with her finger 502.” *Id.*, ¶[0074]. Specifically, “[t]he user touches the touch screen 408 at

² There is a typographical error in EX1005 such that this heading is not spaced from paragraph [0079].

the location corresponding to the unlock image 402 which is located initially at the left end of the channel 404.” *Id.* When the user drags the finger “in the direction of movement 504 . . . [t]he unlock image 402 is dragged along the channel 404 as a result of the gesture.” *Id.*, ¶[0075]. When “the user has dragged the unlock image 402 to the right end of the channel 404 . . . [and] releases the unlock image 402 at the right end of the channel 404, the unlock action is complete.” *Id.* Thus, “[u]pon **completion** of the unlock gesture, the device unlocks and displays on the touch screen 408 user-interface objects associated with normal operation of device 400[,]” such as menu 506. *Id.*, ¶[0076].



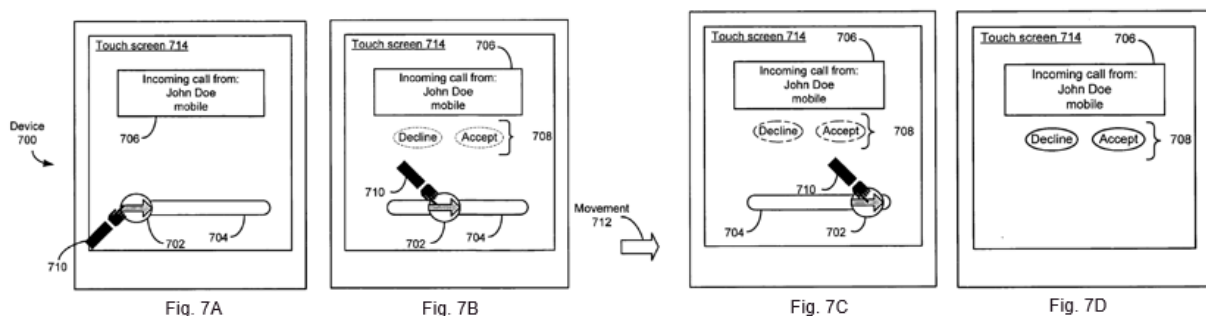
EX1005, FIGs. 5A-5D. A POSITA would have understood that this transition—from a first interface (FIGs. 5A-5C) to a second interface (FIG. 5D)—is performed **after** the gesture is completed. See EX1005, ¶¶[0076]-[0077] (“Upon completion of the unlock gesture,” the device displays the second user interface and failure to complete the gesture results in the device remaining on the lock screen).

59. Chaudhri-842 also describes a second operation for this embodiment, in which a user transitions between two different applications (e.g., between “a telephone application to a music player”), as opposed to a lock and unlock state. EX1005, ¶[0079]. In this application transition scenario, “as the user transitions from a first application and to a second application, a user interface for the second application may fade in (i.e., increase in intensity) and a user interface for the first application may fade out (i.e., decrease in intensity).” EX1005, ¶[0079]. Similar to the lock/unlock scenario, the transition does not occur until *after* the gesture is completed. See EX1005, ¶¶[0076]-[0077]. Chaudhri-842 only describes this fade in/fade out operation for the scenario in which a user transitions between two different applications. EX1005, ¶¶[0052]-[0079]). Thus, Chaudhri-842 discloses two ways to transition between interfaces states, each occurring *after* a condition (e.g., gesture) is completed. See EX1005, ¶¶[0076]-[0079]; see also *id.*, FIG. 3.

2. Indication of Progress Towards Satisfaction of a User Input Condition

60. In this operation, Chaudhri-842 discloses a process for displaying additional interface objects on the lock screen. Here, Chaudhri-842 describes a method of unlocking a device using a predefined gesture via displaying additional interface objects on a lock screen. See EX1005, ¶¶[0091]-[0093]. As shown in FIGs. 7A-7D (inset below), this operation starts when “device 700 is locked . . . [and] display[s] a prompt 706 to the user, informing the user of [an] incoming call .

...” *Id.*, ¶[0091]. When the user “drag[s] unlock image 702 along the channel 704 in the direction of movement 712 . . . a set of virtual buttons 708 appears and increases in optical intensity.” *Id.*, ¶[0092]. When “the user completes the unlock action by dragging the unlock image to the right end of channel 704 . . . [t]he unlock image 702 and the channel 704 disappear from the display and the virtual buttons 708 are at their final optical intensity levels.” *Id.*, ¶[0093]. Once the virtual buttons are at their final intensity levels, “the user may interact with virtual buttons 708 and accept or decline the incoming call.” *Id.*



EX1005, FIGs. 7A-7D.

61. Similar to the operation above (*supra*, §VIII.B.1), the device only transitions “to the second user-interface state *if* the condition is satisfied.” EX1005, ¶[0084]. That is, the transition occurs only *after* the gesture is completed. As shown in FIGs. 7C-7D, the indicator (e.g., image 702 and channel 704) do not disappear or fade out until *after* the gesture is determined to be completed. EX1005, ¶[0093] (“the user completes the unlock action by dragging the unlock image to the right end of the channel 704 and releasing the unlock image 702. The device 700 transitions

to the unlock state. The unlock image 702 and the channel 704 disappear from the display and the virtual buttons 708 are at their final optical intensity levels, as illustrated by their solid outlines.”).

IX. GROUND 1: DR. TERVEEN’S ANALYSIS FAILS TO SHOW OBVIOUSNESS OF CLAIM 1 BY AHN AND CHAUDHRI-842

62. Dr. Terveen’s analysis of Ahn and Chaudhri-842 fails to establish that his proposed combination would arrive at each of the claimed limitations.

A. Limitation [1.d]

63. Limitation [1.d.i] recites that “*during* detection of at least a portion of the gesture before a completion thereof is detected” the processor executes instructions to cause the device to “blur [at least a portion of the at least one other object] based on a change in a magnitude of the gesture.” *Supra* VII.A. That is, Limitation [1.d.i] requires to (1) blur the at least one other object based on a change in a magnitude of the gesture and (2) that the blur occurs “during detection of” the gesture and “*before* a completion” of the gesture is detected. EX1001, 102:41-46. Limitation [1.d.ii] recites that the magnitude of the blur “is increased as a function of an increase in the distance magnitude” of the gesture. EX1001, 102:41-50. Thus, Limitation [1.d.ii] adds that (3) the blur of the at least one other object increases as a function of the distance magnitude of the gesture.

64. “[T]he at least one other object” is introduced in Limitation [1.b], which recites “display an object and at least one other object.” For Limitation [1.b], Dr.

Terveen relies on the “clock widget of [Ahn’s] standby screen 300” as the claimed “at least one other object.” EX1003, ¶¶64-65 (same). Dr. Terveen does not identify any other elements as allegedly corresponding to the claimed “at least one other object.” Thus, for Limitations [1.d.i]-[1.d.ii], Dr. Terveen must show that Ahn-Chaudhri-842 would have rendered obvious blurring Ahn’s “clock widget” as required by the language of the claim. But Dr. Terveen fails to demonstrate this point.

1. Dr. Terveen fails to Show Ahn’s Graphical Shadow Moves

65. Limitation [1.d.i] recites “based on a change in a distance magnitude of the gesture being detected on the touch screen *to thereby move the object* on the touch screen.” Dr. Terveen relies solely on paragraph [0118] of Ahn to support that “graphic 1800 moves (i.e., is ‘dragged’) in response to and along with a user’s touch and drag gesture.” EX1003, ¶78. But paragraph [0118] of Ahn does not describe the graphic shadow 1800 as moving. Further, Dr. Terveen’s cites to Chaudhri ’842’s discussion of “cues,” but does so solely to suggest an interpretation of Ahn. EX1003, ¶79. Notably, the Petition does not articulate any modification of Ahn’s graphic shadow 1800. Pet., 40-41.

66. Dr. Terveen’s proposed mapping rests on the assumption that the phrase “*dragged* graphic 1800” (EX1004, ¶[0118]) means that the graphic is moved. EX1003, ¶78. But this assumption is inconsistent with Ahn’s use of the term

“dragged,” which is used to describe the input and not the movement of UI icons. In Ahn, the term drag explicitly “indicates movement of an external object on the touch screen 10.” EX1004, ¶[0098]. In other words, drag means an external object (e.g., finger, stylus, etc.) is acting on the interface, not that any object is moving in the interface. For example, Ahn states that “if the user drags *the touch screen 10* by a distance longer than or equal to the distance d1 and shorter than the distance d2, the first menu list 1700 is displayed.” EX1004, ¶[0116]. Of course, the user is not dragging the actual touch screen 10 (*see* EX1004, FIG. 1) but is merely performing a drag gesture on the touch screen, in response to which the first menu list is displayed. EX1004, ¶[0102] (“drags the touch”).

67. Similarly, in the context of Ahn’s paragraph [0118], the user is not dragging the graphic shadow 1800, but merely performing a drag gesture on the graphic. This understand is consistent with Ahn’s explanation that graphical shadow 1800 is specifically “for notifying a *start* position of a touch and drag operation.” EX1004, ¶[0118]. Graphic 1800 merely notifies a user of where to begin (i.e., start) the touch and drag gesture; Ahn does not disclose or suggest that the graphic 1800 moves with the drag. *See* EX1004, FIG. 18(a) (depicting graphic 1800 remaining stationary as a finger is depicted as dragging across the screen).

68. Dr. Terveen provides no explanation or reasoning as to why “drag” in Ahn’s paragraph [0118] would be read in a different context than is used in the rest

of Ahn's description. See EX1004, ¶¶[0098], [0116]. Instead, Dr. Terveen makes an unexplained assumption that is inconsistent with Ahn's use of the term dragged. Further, Dr. Terveen provides no explanation for why moving of the shadow, which Ahn explicitly discloses as indicating "a *start position* of a touch and drag operation," would have served any purpose. EX1004, ¶[0118]. Because Dr. Terveen maps exclusively to the graphical shadow 1800 as the claimed "object" (EX1003, ¶¶65, 78), he cannot show that Ahn's gesture is "to thereby move the object on the touch screen" as stated in Limitation [1.d.1]. For this reason, Dr. Terveen's proposed combination fails to teach each and every feature of claim 1.

2. Ahn-Chaudhri-842 would not increase magnitude of the change in appearance of Ahn's clock widget as a function of an increase in the distance magnitude of the gesture

69. As explained above, for Limitation [1.d.ii] Dr. Terveen must show that "blur" of the clock widget increases as a function of the distance magnitude of the gesture. Dr. Terveen's analysis of Limitation [1.d.ii] focuses solely on the portion of the clock widget being blurred, not that the blur is increased as a function of the gesture. EX1003, ¶¶81-83. Ahn fails to describe any gradual change in the appearance of its clock widget. *see generally* EX1004. And, while Dr. Terveen cites

to paragraph [0079] of Chaudhri-842 as teaching that a first application³ may “fade out,” this is not done as a function of an increase in the distance magnitude. Instead, Chaudhri-842 explains this is performed according to a pre-set *temporal* interval, not a distance magnitude. EX1005, ¶[0079] (“over a *pre-determined* time interval”). Thus, the only description of decreasing an optical intensity in Chaudhri-842 is performed as a function of a pre-determined time interval, not as a function of an increase in the distance magnitude. As Ahn fails to describe any decreasing an optical intensity of its clock widget, there is no record evidence to support the combination would arrive at the claimed features.

70. Further, Dr. Terveen’s incorporation of its analysis from Limitation [1.d.i] cannot support Dr. Terveen’s proposed mapping regarding Limitation [1.d.ii] because neither Ahn nor Chaudhri-842 discloses decreasing the optical intensity of a background object during a gesture. *Supra*, §IX.A.3.c). At no point does the magnitude of the optical intensity of the first application UI decrease as a function

³ As explained in Section VIII.A.3.c)(2), switching between the UIs of two distinct *applications* is not readily transferable to Dr. Terveen’s proposed modification. When the user transitions from a first application and to a second application the entire first application UI must be removed from the display; whereas Ahn’s background remains on the display.

of an increase in the distance magnitude of the gesture. For at least this reason, Dr. Terveen's proposed mapping for Limitation [1.d] must fail.

3. *Ahn-Chaudhri-842 would not blur during detection of the gesture and before a completion of the gesture is detected*

71. Dr. Terveen asserts that the change in the appearance of Ahn's clock widget (EX1004, FIG. 4B) discloses the claimed "blurring"⁴ by applying the use of transitioning optical intensity of Chaudhri-842. EX1003, ¶¶71-72. "The optical intensity of a user-interface object, as used herein, is the object's degree of visual materialization," (EX1005, [0085]), which Dr. Terveen identifies as "transparency" or translucency. EX1003, ¶74 (emphasis in original); *see also id.*, ¶76 ("utilizing this progressive translucency or blurring effect"). However, Dr. Terveen's mapping is erroneous for at least three reasons: (1) none of the cited references disclose decreasing translucency *during*—and before completion of—a gesture based on a distance magnitude of the gesture; (2) the proposed combination is inconsistent with the teachings of Chaudhri-842 and is only navigable by using the claims as a roadmap; and (3) there is no motivation to make the proposed modifications between Ahn's embodiments and Chaudhri's teachings.

⁴I do not agree with Dr. Terveen's proposed interpretation of blurring. Nevertheless, Dr. Terveen's analysis is erroneous even under Dr. Terveen's own interpretation.

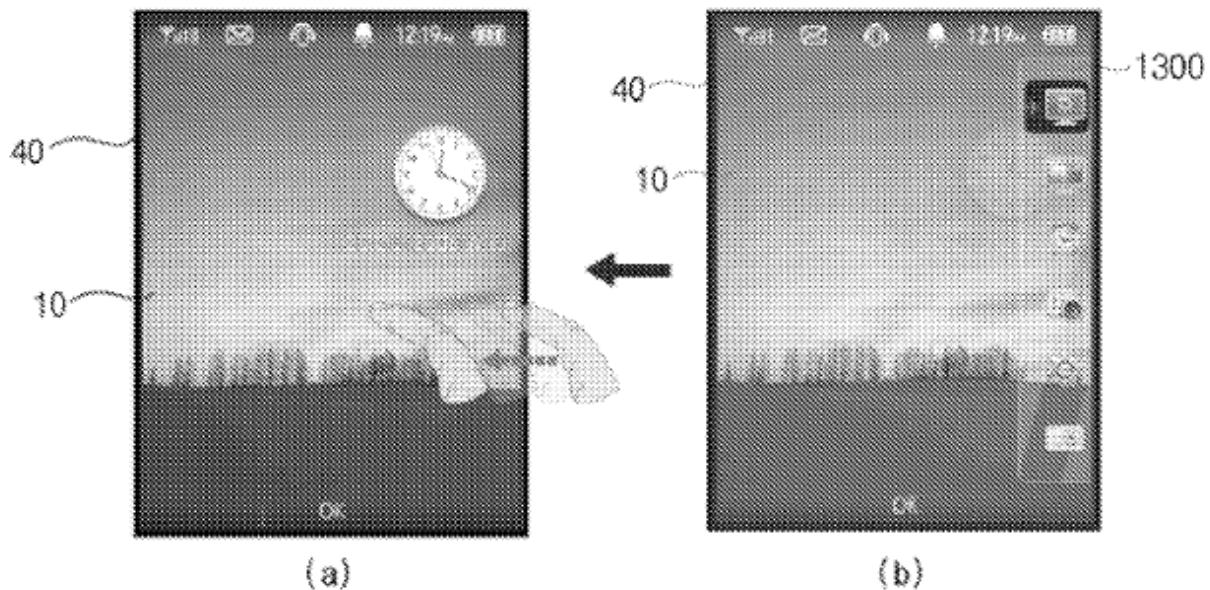
72. Dr. Terveen fails to show that display of Ahn’s menu 420 would cause the alleged ‘blurring’ of the clock widget “during detection of at least a portion of the gesture” and “before a completion [of the gesture] is detected.” At no point does Ahn explain, identify, or even allude to the change in appearance of its clock widget. *See generally*, EX1004. In fact, the only discussion in Ahn of translucency is in reference to menus, and not the background standby screen or the clock widget. *See* EX1004, ¶[0070]. As such, Dr. Terveen relies exclusively on the figures of Ahn for the “blurring” of the clock widget. *See e.g.*, EX1003, ¶71. But this fails to establish that Ahn’s clock is “blurred” *during* the gesture in Dr. Terveen’s mapping. Further, Chaudhri-842 does not suggest either of Dr. Terveen’s modifications of changing the appearance of a background *during* detection of a gesture or *decreasing* optical intensity of a background that remains visible upon completion of the gesture. Instead, Chaudhri-842 *increases* the optical intensity of a *foreground* element, or entirely fading out a previous application when transitioning to a new application.

a) Ahn’s embodiments do not blur before a completion of the gesture is detected

73. Dr. Terveen’s analysis of Limitation [1.c] maps the claimed “gesture” to a “touch and drag operation” associated with opening Ahn’s menu list 1300, which matches Ahn’s user menu 420. EX1003, ¶¶67-68; *compare* EX1004, FIG. 13 *with id.*, FIG. 4B. There is no evidence that Ahn’s clock widget is “blurred” during this gesture.

74. First, referring to Ahn's "touch and drag" embodiment, shown in FIGs. 13A-B (inset below), a user touches the screen and "drags the touch in a left direction" (EX1004, ¶[0102]) and *afterwards* a menu list 1300 matched to a left direction is displayed (FIG. 13B).

FIG. 13

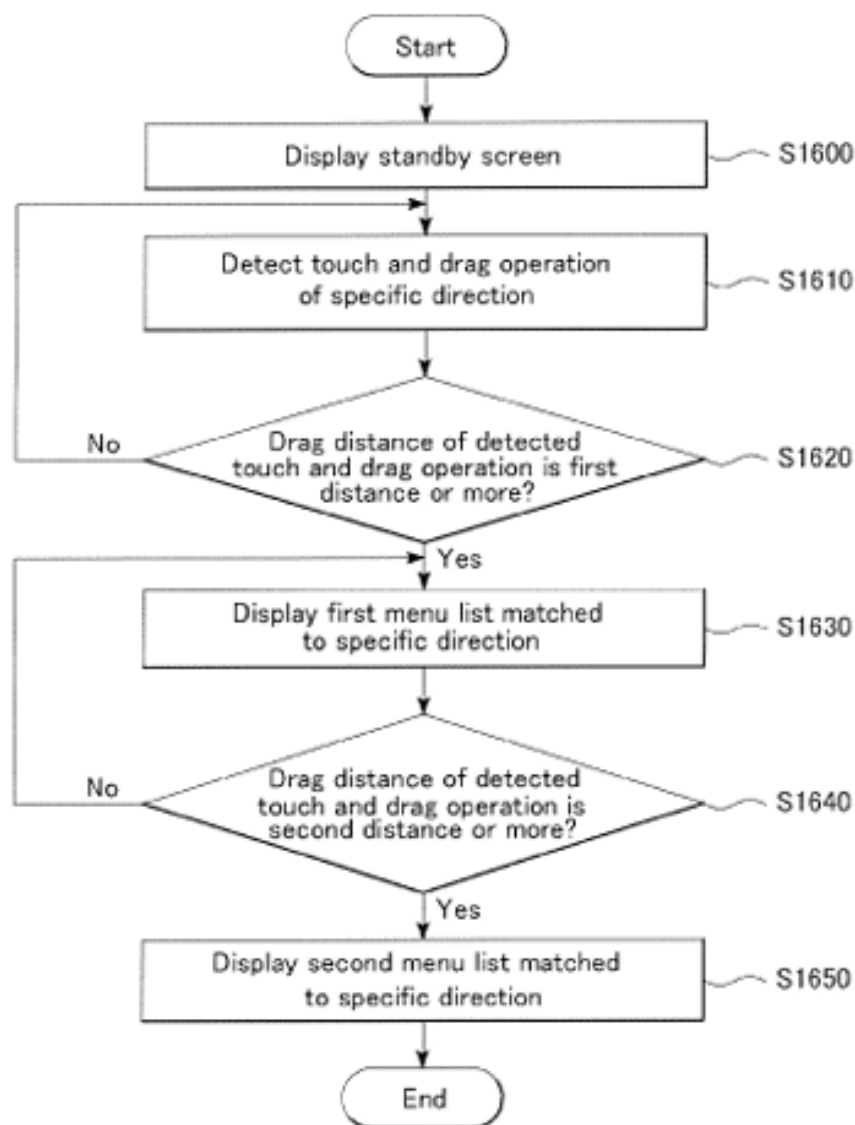


EX2005, 71/75.⁵

75. As shown in FIG. 13(a), the user's finger drags from a first point to a second point on the display, while the display does not change as the user's finger

⁵ For improved clarity, the figure is from the counterpart European application (EP1962480) that is substantially identical to Ahn and claims priority to the same applications as Ahn. See EX2005, 78/79 (claiming priority to KR 2007-018175, KR 2007-033308, KR 2007-043087); EX1004, 1 (same).

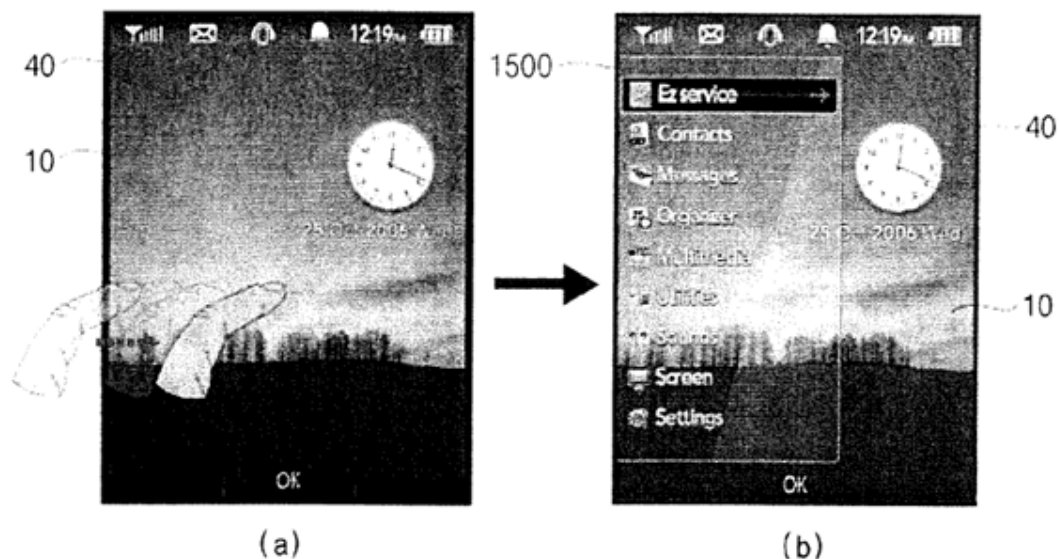
“drags the touch in a left direction.” See EX1004, ¶[0102]. Only *after* “the touch and drag operation [i.e., gesture] of the user is completed” is the appearance of the clock changed (FIG. 13b). See EX1004, ¶[0116]. This operation is confirmed by Ahn’s FIG. 16 (inset below), in which Ahn displays its menu (1630) only *after* determining a drag distance exceeded a first distance (1620). See EX1004, ¶[0107]-[0109].



EX1004, FIG. 16. “That is, if the touch and drag operation of the user *is completed*, the controller 50 determines a distance of the touch and drag operation, whereby a specific menu list corresponding to the distance can be displayed in the display 40.” EX1004, ¶[0116]. Thus, in Ahn’s touch and drag embodiments, the menu 1300 is not displayed *during* the gesture; rather, it is displayed only *after* the gesture is completed.

76. Second, referring to Ahn’s “Start Position” embodiments, a graphical shadow 1800 is utilized to display menu list 1500. As shown in Ahn’s FIGs. 15a-15b (inset below), the display of menu list 1500 does not dim, obstruct, or otherwise alter the appearance of the clock widget.

FIG. 15

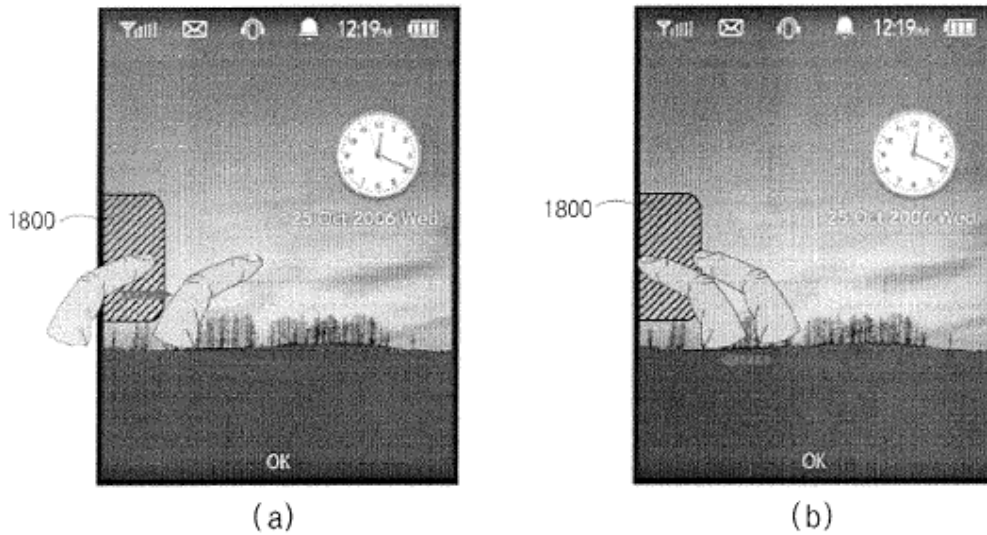


EX1004, FIG. 15A-B (appearance of clock widget is not changed from FIG 15A to FIG. 15B) *see also* FIGs. 8C, 9A, 9B (same).

77. Contrary to the “Touch and Drag” operation of FIGs. 13-15 where a menu list is displayed according to a specific direction of a gesture (EX1004, ¶[0101]), the “Start Position” provides a method of opening a single menu using two different gestures. *Supra*, §VIII.A.3. However, neither of the two functions of the “Start Position” embodiment changes the appearance of the clock widget. Ahn’s “FIG. 18 illustrates *an example* of a screen for displaying a graphical display for notifying a start position of a touch and drag operation to a user.” EX1004, [0118]; *see also id.*, [0025]. Thus, Figures 18A and 18B illustrate examples of functions of a single common embodiment.

78. In the first function, illustrated in FIG. 18A (shown below), “if the user *drags in a right direction* after touching the screen, a graphical shadow 1800 is displayed” and then, after the drag in the right direction is completed, the user may “drag” the graphical shadow 1800 (i.e., perform a touch and drag operation starting on the graphical shadow) to display menu list 1500. EX1004, ¶[0118].

FIG. 18

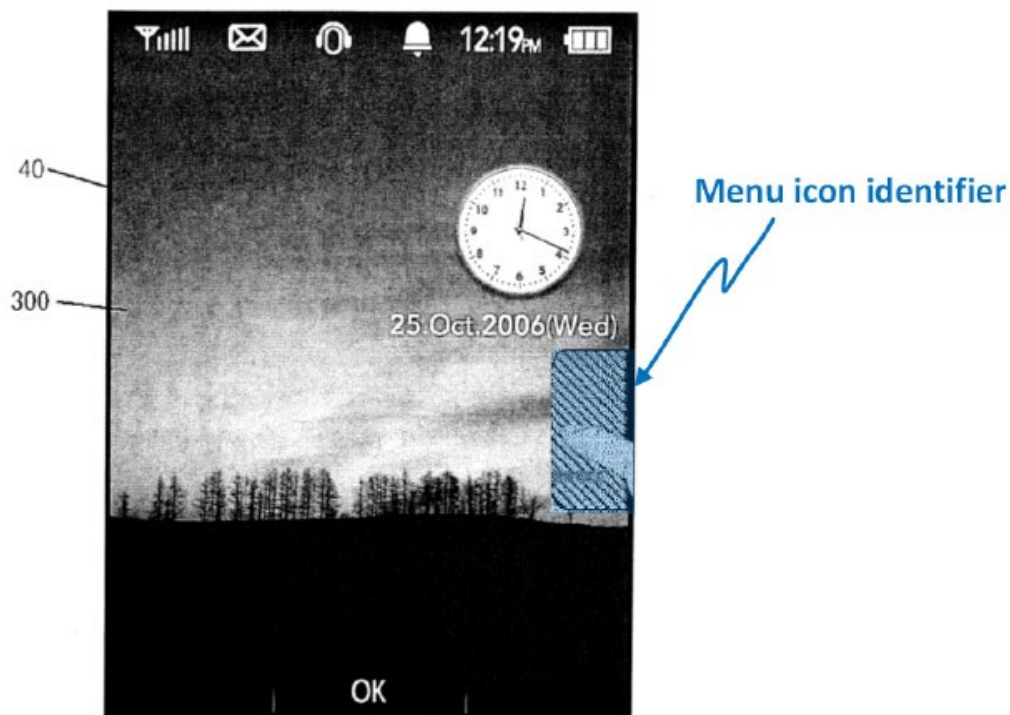


EX1004, FIGs. 18A-B. There is nothing in Ahn that suggests that any object is “blurred” during the initial touch and drag in a right direction or while the user’s finger “drags” the graphical shadow 1800 .

79. In the second function, illustrated in FIG. 18B (shown above), the user performs a touch and ***drag in a left direction*** to display a graphical shadow 1800. EX1004, ¶[0119]. The user may then “press the graphic 1800 with a finger” to cause the graphic 1800 gradually disappears in the display 40 and the menu list 1500 can be displayed. EX1004, ¶[0119]. As with the first embodiment, there is nothing in Ahn that suggests that any object is “blurred” ***during*** the touch and drag in a left direction. Thus, Ahn fails to disclose “blurring” any object based on an interaction with graphical shadow 1800.

b) Dr. Terveen's Amalgamation of Ahn does not blur before a completion of the gesture is detected

80. As none of the embodiments disclosed in Ahn teach the claimed feature, Dr. Terveen must modify Ahn internally and with Chaudhri-842 to arrive at claim 1. Dr. Terveen opines that a POSITA “found obvious that the menu identifier icons would have been associated with any of the menus, including, for example, the user menu illustrated in FIGS. 4B and 13b” and presents the below “amalgamation” of Ahn. EX1003, ¶63.



EX1004, Amalgamation of FIGS. 3A and 18a

EX1003, ¶¶63-64.

81. Dr. Terveen does not explain why his proposed amalgamation would “blur” the clock widget *during* a gesture to move graphical shadow 1800. Dr.

Terveen cites to no evidence to support that the proposed amalgamation would operate contrary to Ahn’s “Touch and Drag” embodiment—which changes the appearance of the clock widget *after* the completion of the gesture (*Supra*, §IX.A.3.a). EX1003, ¶72 (tacitly conceding “Ahn does not specify ... the manner in which a POSITA would have implemented the above-described blurring of its clock widget”).

82. Dr. Terveen merely cites to Ahn’s FIGs. 4B, 10, and 13B as support that Ahn illustrates blurring at least a portion of background standby screen 300 (e.g., a clock widget) as a result of a user’s gesture opening one of the menus. EX1003, ¶71. But this fails to establish that Ahn’s clock is “blurred” *during* the gesture in Dr. Terveen’s proposed amalgamation. Thus, to the extent Dr. Terveen intends to assert⁶ that Ahn’s clock widget would change appearance while graphical shadow 1800 is being “dragged,” this assumption is unsupported and contradicted by the operation of Ahn.

⁶ Dr. Terveen does not explicitly make this assertion in his declaration. *See* EX1003, ¶¶71-72.

c) Dr. Terveen's Ahn-Chaudhri Modification does not blur before a completion of the gesture is detected

(1) The cited portion of Chaudhri-842 does not support Dr. Terveen's Modification

83. Dr. Terveen points to Chaudhri-842 as remedying the deficiencies of Ahn and broadly opines that “a POSITA would have found it obvious to apply Chaudhri '842's user interface state transition techniques—particularly the use of transitioning optical intensity to provide visual feedback regarding the progress of a transition between user interface states as a result of detecting a user's input of a predefined gesture—in the context of Ahn's mobile communication terminal 100.” EX1003, ¶72. But Chaudhri-842 fails to disclose “blurring” a standby screen or background object during a gesture.

84. Dr. Terveen's analysis of Limitation [1.d.i] relies on Chaudhri-842's teaching of *increasing* the optical intensity of *foreground* objects associated with a *second* user interface. EX1003, ¶¶72-77 (citing EX1005, ¶¶[0080]-[0089] and FIG. 6). This teaching does not include decreasing the optical intensity of the background of the first user interface. EX1005, ¶¶[0080]-[0082] (“progress is indicated . . . by transitioning optical intensity of one or more user interface objects associated with the second user-interface state”), [0092]-[0093] (“[a]s the user drags the unlock image, a set of virtual buttons 708 appears and increases in optical intensity”). But, as explained in Section IX.A.4.b) below, this teaching is not commensurate with Dr.

Terveen’s proposed modification to *decrease* a translucency of a *background* object (clock widget) that is part of the *first* user interface state. *Compare* Pet., EX1003, ¶¶71-79 (proposing to *decrease* appearance of *background* clock widget during gesture); *with* EX1005, ¶¶[0091]-[0093] (describing *increasing* optical intensity of *foreground* buttons 708 during gesture, but only *decreasing* optical intensity of image 702 and channel 704 *after* the gesture is completed—“by dragging the unlock image to the right end of the channel 704 and *releasing* the unlock image 702.”).

85. Dr. Terveen cites to nothing in Chaudhri-842 that supports the required modification to *decrease* the optical intensity of Ahn’s background object *during*—and as a function of—a user’s gesture. To the extent Dr. Terveen is attempting to read the term “transitioning an optical intensity” broadly to include decreasing the optical intensity, this is incorrect. Chaudhri-842 only teaches “transitioning an optical intensity of one or more user interface objects *associated with the second user-interface state*—*i.e.*, increasing optical intensity of an emerging foreground object—not decreasing optical intensity of a remaining background. EX1005, ¶[0082]. As the second user-interface state is not displayed, a POSITA would understand that they cannot decrease the optical intensity of an object that is not visible. And the cited operation of Chaudhri-842 does not extend this transition in optical intensity to any object associated with the first user-interface state. EX1005, ¶[0080]-[0098]. With nothing to bridge the gap between Chaudhri-842’s teaching

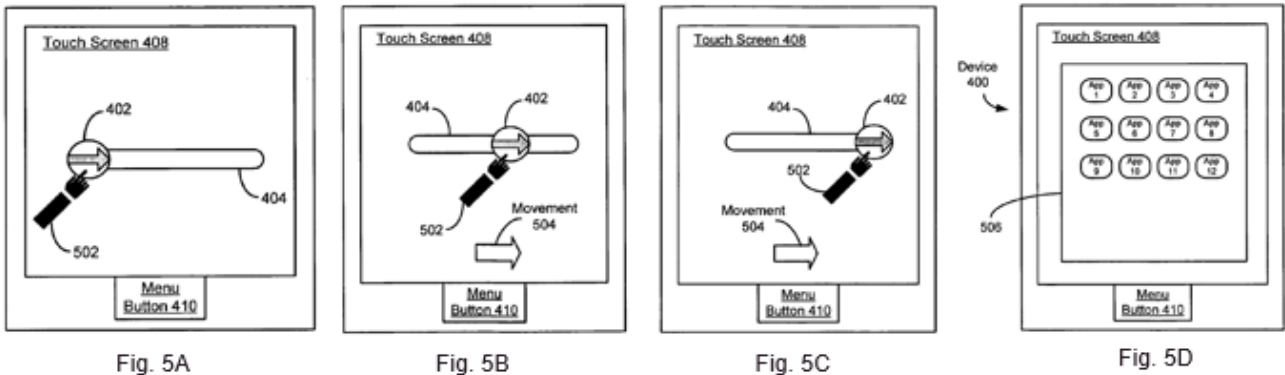
to increase optical intensity of a foreground object in a second user interface and Dr. Terveen's proposed modification to decrease optical intensity of a background object in a first user interface, Dr. Terveen's proposed mapping for Limitation [1.d] must fail.

(2) The other operations of Chaudhri-842 do not support Dr. Terveen's Modification

86. The only citation relevant to Dr. Terveen's proposed modification to *decrease* the optical intensity of Ahn's clock is found in his declaration's analysis of another Limitation ([1.d.ii]), which cites to a separate section of Chaudhri-842. EX1003, ¶¶81-83 (citing EX1005, ¶[0079]); EX1005, ¶¶[0052]-[0079] (Under heading "Unlocking a Device via Gestures"); EX1005, ¶¶[0080]-[0098] (Under heading "Indication of Progress Towards Satisfaction of a User Input Condition"). However, the cited portions of Chaudhri-842 do not support Dr. Terveen's proposed modification.

87. Specifically, Chaudhri-842's disclosure of fading out in the "unlocking a Device via Gestures" section fails to support Dr. Terveen's modification. In this section, Chaudhri-842 provides two examples of this transition: (1) between a lock state and an unlock state (EX1005, ¶¶[0048], [0052]-[0078]) and (2) between a first and second application (EX1005, ¶¶[0048], [0079]).

88. As shown in Chaudhri-842's FIGs. 5A-5D (inset below), a user may perform the unlocking function via a gesture—by touching image 402 and sliding to the end of channel 404. EX1005, ¶¶[0074]-[0076].



Once the user completes the gesture—“release[ing] the unlock image 402 at the right end of the channel 404”—the gesture (i.e., unlock action) is complete. EX1005, ¶[0076]. “*Upon completion* of the unlock gesture, the device unlocks and displays on the touch screen 408 user-interface objects associated with normal operation of the device 400.” *Id.* In other words, *after “completion* of the unlock gesture,” the device transitions from a first user-interface (lock screen in FIG. 5C disappears from display) to a second user-interface (menu 506 from standby screen in FIG. 5D appears on display).

89. Paragraph [0079] of Chaudhri-842 describes an example where the device transitions from one application to another, such as “from a telephone application to a music player or vice versa.” EX1005, ¶[0079]. Chaudhri-842 explains that “as the user transitions from a first application and to a second

application, a user interface for the second application may fade in (i.e., increase in intensity) and a user interface for the first application may fade out (i.e., decrease in intensity).” *Id.* Chaudhri-842 does not disclose changing the timing of this fade in/fade out feature from lock/unlock operation and a POSITA would have understood that the first application UI will fade out *after* completion of the unlock gesture. *See id.* That is, the completion of the gesture will start the transition between applications and “as the user transitions” between the applications the second UI will fade in, and the first UI will fade out. This fade in and fade out can then occur over a fixed, “pre-determined time interval.” EX1005, ¶[0079].

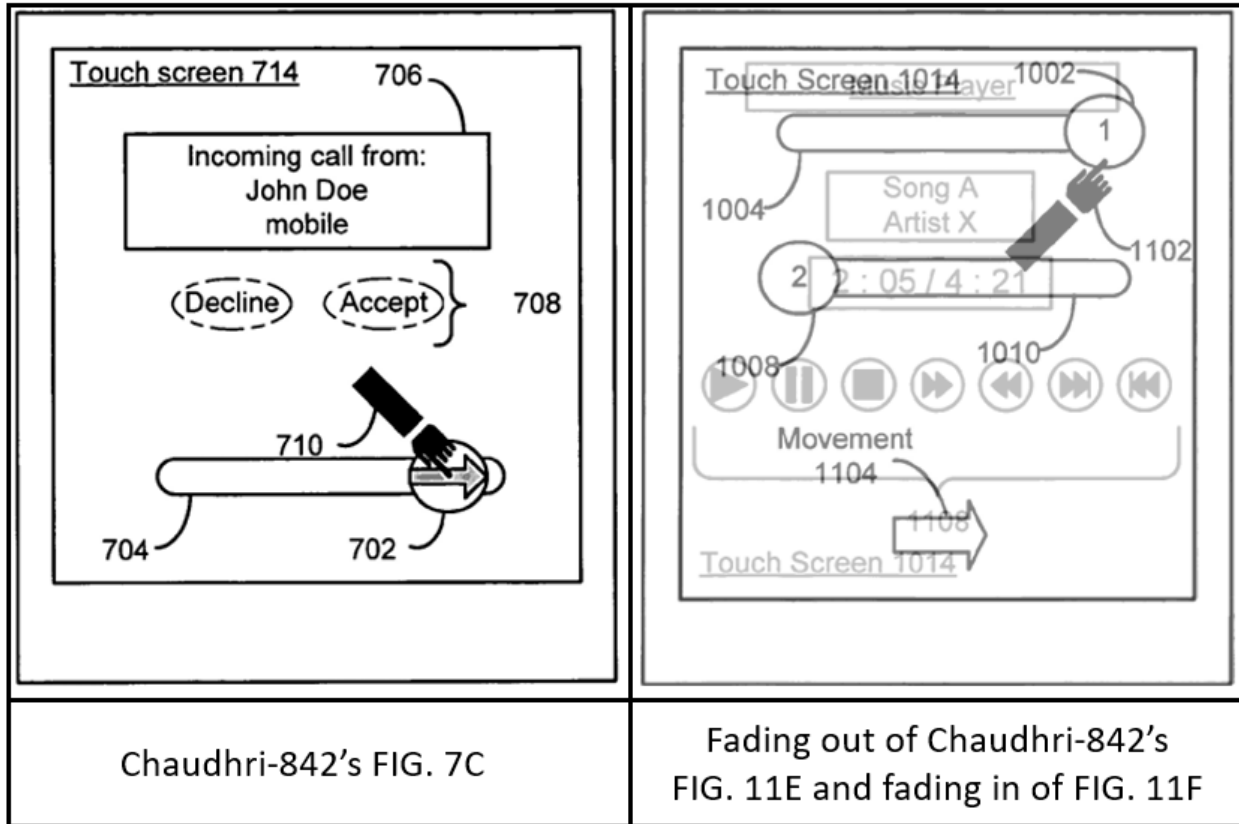
90. The time it takes for a given gesture to be performed by a given user, cannot be determined until after the gesture is completed. As such, the length of Chaudhri-842 transition cannot be “pre-determined” until after the gesture has been completed. A POSITA therefore would have understood the reference to “[t]he pre-determined time” associated with the unlock gesture would not occur *during* the unlock gesture. Following this, a POSITA would have understood this transition to occur *after* completion of the gesture as illustrated by the figures of Chaudhri-842. EX1004; ¶[0076] (“upon completion”), [0093] (ser completes the unlock action), [0106] (“upon completion of the unlock action”); *see also* EX1004, FIG. 5C to FIG. 5D; FIG. 11C to FIG. 11D; and FIG. 11E to FIG. 11F (showing the objects of the first interface state disappear *after* the gesture is complete). Therefore, applying this

teaching to Ahn's embodiments would have simply resulted in the same operation as disclosed by Ahn, where the appearance of the clock is changed *after* the gesture is completed—not during the gesture, before a completion thereof.

91. Further, the fading out of the *entire* user interface for the first application, and the fading in of the *entire* user interface for the second application, does not teach and is not applicable to Dr. Terveen's proposed modification to decrease the optical intensity of an isolated background object. Unlike Ahn's menu opening operation—where a menu is overlaid on a portion of the background—the transition between applications requires the display to completely change interfaces. This can only be done by ceasing the display of the first application UI and transitioning to display the second application UI. The fact that Chaudhri-842 does this transition by fading in and fading out the UIs does not explain why a POSITA would fade out Ahn's background clock widget. Ahn's transition is not between two applications because Ahn's standby screen and menu 420 are not applications. *See* EX1004, ¶¶[0063]-[0064], FIG. 7B; *see also* EX1005, ¶[0079]). Nor does Ahn replace the entire background display. EX1004, FIG. 13 (displaying background and menu). In fact, Ahn's clock widget is still visible when the menu appears (EX1004, FIG. 4B); whereas Chaudhri-842's first application fades out completely.

92. Ahn displays a background screen and then displays a popout menu over a portion of that background screen. This does not necessitate ceasing the

display of the background screen and, in fact, Ahn chooses to continue displaying the background screen. Ahn's and Chaudhri-842's goals are completely opposing on this point. Ahn emphasizes the benefits of its directional pop out menus is to enable the standby screen to *remain visible* while using the menus. *Supra*, §VIII.A; EX1004, ¶¶[0005]-[0006], [0039], [0064] (displaying menus and standby screen concurrently). On the other hand, a POSITA would have understood that allowing the first application UI to *remain visible* during a transition from a first application to a second application would lead to a poor user experience. For example, transitioning between two different application UIs would likely result in a large overlap of the interfaces during fading, which would lead to user confusion. This can be seen in the below figures. Unlike the isolated fading in of two buttons as shown in Chaudhri-842's FIG. 7C, the application UI takes up much more space and would cause significant overlap in the display, as shown by the combination of Chaudhri-842's FIG. 11E (unlock screen) and FIG. 11F ("music player application"). EX1005, ¶¶[0079], [0107].

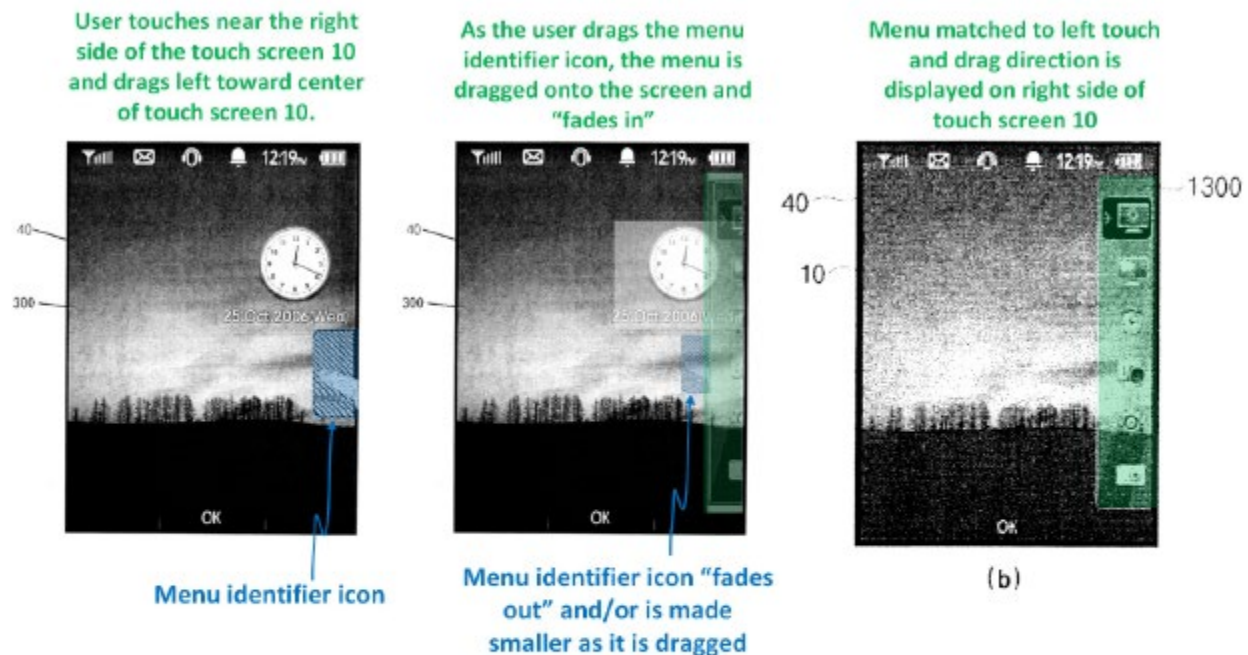


As shown, the image combining Chaudhri-842's FIG. 11E and FIG. 11F (both set at 50% transparency) would replicate what the user would see at 50% progress of the unlock gesture (EX1005, ¶[0087]), but this fails to provide a clear indication of progress as the significant overlap of UI elements leads to a poor user experience. If the user were manipulating the unlock image during this transition, it would lead to confusion that may prevent the user from completing the gesture; whereas if this transition occurred after the user knowingly completed the unlock gesture, the user would understand the transition is a result of the completion of the gesture. Thus, a POSITA would understand Chaudhri-842's fade in/fade out of applications as occurring after the gesture is complete.

93. Further, Chaudhri-842 does not extend this fade in/fade out to transitions between a lock screen and a menu (EX1005, ¶¶[0074]-[0078]) and does not teach to fade out a single object associated with the first user-interface (EX1005, ¶¶[0080]-[0094]). Thus, Dr. Terveen's proposed modification to decrease the optical intensity of Ahn's clock widget when transitioning from a standby screen to a popout menu is not commensurate with the teachings of Chaudhri-842.

4. Dr. Terveen's Proposed Ahn-Chaudhri Combination does not flow from the teachings of Ahn or Chaudhri-842

94. Dr. Terveen's proposed modification of Ahn with Chaudhri-842 is only logical when using the claims of the '727 Patent as a roadmap to reconstruct the claimed invention. Starting with Dr. Terveen's "amalgamation" of Ahn, graphical shadow 1800 is used to denote the start position for a gesture to bring out the user menu 1300 illustrated in FIG. 13b. EX1003, ¶¶63-64. Dr. Terveen then seeks to modify Ahn's graphical shadow 1800 operation so that the menu 1300 "fades in" as it is dragged onto the screen and the graphical shadow 1800 "fades out or is made smaller as it is dragged. EX1003, ¶¶78-79. This modified operation is shown in the figure below.



EX1004, Amalgamation of FIGS. 3A and 18a (left), Amalgamation of FIGS. 3A, 4A, and 18a (center), and FIG. 13b (right)

Id.

95. This operation is only possible by conflating Ahn's two separate 'Start Position Operation' embodiments. Ahn specifically provides an option where the user may press the graphic 1800 and after the gesture (i.e., press) is completed, "the graphic 1800 gradually disappears in the display 40, and the menu list 1500 can be displayed as emerging from the ... display 40."⁷ EX1004, ¶[0119]. But this fails to meet the claimed operation, as the gradual disappearance does not occur during the

⁷ There is no embodiment of Ahn describing the Ahn's clock widget as disappearing in the same manner as the graphic 1800.

gesture or increase as a function of a distance magnitude. Thus, Dr. Terveen seeks to merge the two functions of Ahn's Start Position Operation embodiments. *Supra*, §VIII.A.3 Dr. Terveen provides no reasoning or explanation for this intermingling of embodiments and Dr. Terveen's reliance on Chaudhri-842 does not provide the required explanation.

96. Similar to his mixing of Ahn's embodiments, Dr. Terveen must blend two of Chaudhri-842's operations to arrive at the modification that he proposes. Again, Dr. Terveen fails to provide any reasoning or explanation for this mixing and matching. First, Dr. Terveen relies on Chaudhri-842's "Indication of Progress Towards Satisfaction of a User Input Condition" operation, which teaches increasing the optical intensity of an object of a second user-interface. But this fails to disclose decreasing the optical intensity of a background object of a first user-interface. Second, Dr. Terveen relies on Chaudhri-842's "Unlocking a Device via Gestures" operation, which teaches fading in an entire user-interface of a second application and fading out an entire user-interface of a first application. But this fails to disclose decreasing the optical intensity of a single background object of a first user-interface. Thus, Dr. Terveen seeks to merge Chaudhri-842's mutually exclusive "Indication of Progress Towards Satisfaction of a User Input Condition" and "Unlocking a Device via Gestures" operations. *Supra*, §VIII.B, Dr. Terveen provides no reasoning or explanation for combining Chaudhri-842's operations.

97. As explained below, rather than address Ahn's and Chaudhri-842's actual teachings, Dr. Terveen simply uses the challenged claims as a roadmap to select and reassemble certain words of the references to seemingly match what is claimed. However, Dr. Terveen's presentation of Ahn and Chaudhri-842 is incorrect, and motivations offered for modification of Ahn are illogical and would not result in Dr. Terveen's proposed combination.

a) Dr. Terveen's proposed modification is inconsistent with Ahn's described operations

98. Ahn does not disclose fading out graphic shadow 1800, or making the graphical shadow smaller, as it is dragged. As explained above in Section VIII.A.3, Ahn provides two operations of using graphical shadow 1800. In the first embodiment, "as the user drags the graphic shadow 1800, the menu list 1500 can be displayed along the dragged graphic 1800." EX1004, ¶[0118]. In this operation, the menu does not fade in, nor does the graphic shadow 1800 fade out (or become smaller). In Ahn's second operation, the user may press (e.g., tap) the graphic shadow 1800 and "the menu list 1500 can be displayed as emerging from the left side of the display 40 while the graphic 1800 disappears in the display 40." EX1004, ¶[0119]. In this second operation, neither the menu nor the graphic shadow 1800 slide in.

99. Dr. Terveen does not provide any rationale for integrating the teachings of Ahn's mutually exclusive embodiments. *See* EX1003, ¶¶71-79. Nor does the Dr.

Terveen contend that Chaudhri-842 provides a reason to make this modification. *See e.g.*, EX1003, ¶72 (applying “Chaudhri ’842’s user interface state transition techniques” only to the “blurring of [Ahn’s] clock widget”). Nor would this flow from Chaudhri-842’s teachings, which disclose increasing optical intensity of stationary buttons. EX1005, ¶¶[0080]-[0094].

b) Dr. Terveen’s proposed modification is inconsistent with Chaudhri-842’s varying the optical intensity of foreground UI elements associated with a second user-interface state

100. Chaudhri-842 does not vary the optical intensity of a background. To the contrary, Chaudhri-842 discloses adjusting an optical intensity of user interface (UI) objects and repeatedly makes clear that user interface objects are distinct from a background. *See, e.g.*, EX1005 [0085] (“optical intensity may be the visual differentiation *between* the user-interface object and *the background*”). Thus, Chaudhri-842 teaches varying the optical intensity of foreground UI elements.

101. Unlike Chaudhri-842’s foreground UI elements, Ahn’s clock widget is part of the background, as Dr. Terveen concedes. *See* EX1003, ¶¶71 (“that a clock widget is part of the background standby screen 300”), 76 (“the background (e.g., the standby screen and/or clock widget).”). But Chaudhri-842 does not suggest adjusting the optical intensity (or otherwise altering) a background. Dr. Terveen provides no explanation for this gap in logic. Further, Dr. Terveen’s assertion to “vary ... the optical intensity of the [background] standby screen” by “utilizing [a]

progressive translucency” on the background (EX1003, ¶76) is nonsensical. A POSITA would have understood that varying the transparency of the *background* standby screen is pointless because there is nothing below the background (e.g., underlying layers) that would be revealed by altering its transparency.

102. Additionally, Chaudhri-842 does not decrease the optical intensity of objects associated with a first user-interface. As Dr. Terveen notes in his declaration, Chaudhri-842 describes “a process 600 for indicating progress towards satisfaction of a user input condition” in which an electronic device is transitioned between a first user-interface state and a second user-interface state. EX1005, [0080]-[0082]; EX1003, ¶73. The “progress is indicated (604) towards satisfaction of the condition by transitioning an optical intensity of one or more user interface objects *associated with the second user-interface state*,” as illustrated in Chaudhri’s FIG. 6 (inset and annotated below). EX1005, [0080]-[0082]; EX1003, ¶73.

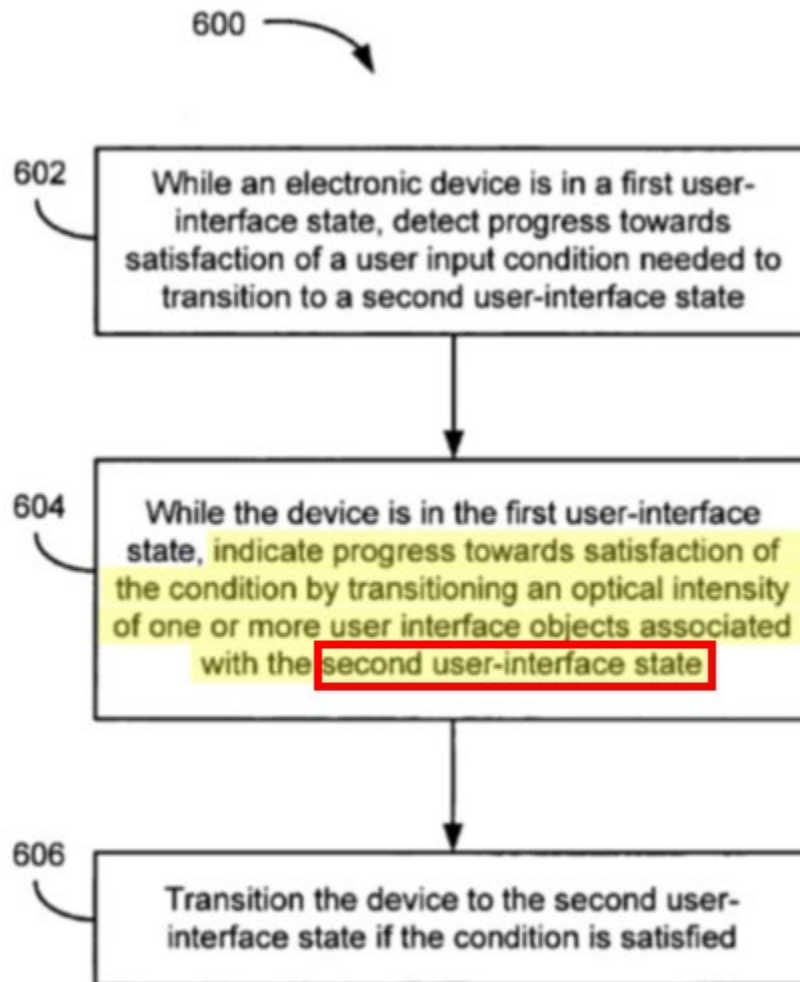
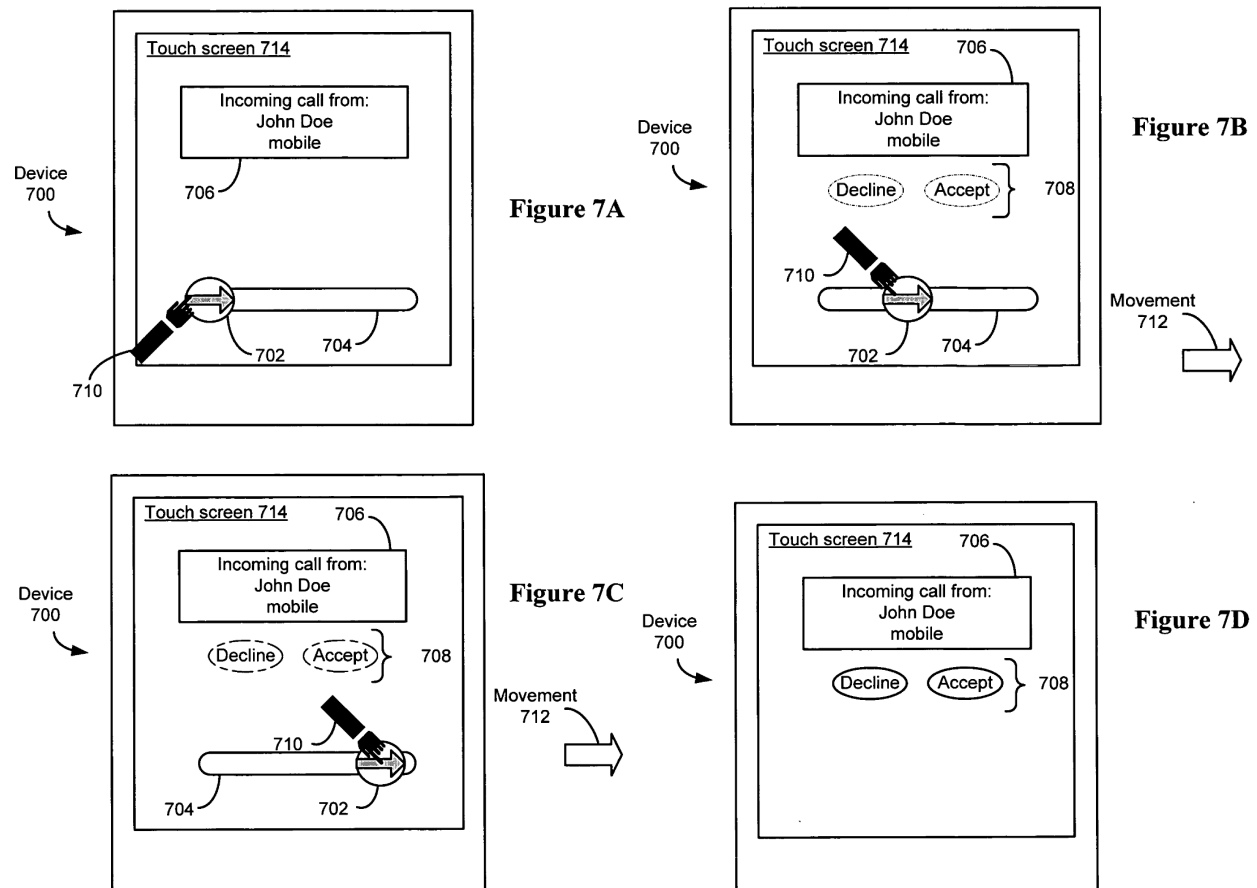


Figure 6

EX1005, FIG. 6.

103. Chaudhri-842 also illustrates an example of this concept in its Figures 7A-7D (inset below). As shown, as the arrow 702 slides to the right, the optical intensity of the virtual buttons 708—associated with the second user interface state—increases. EX1005, [0092]-[0093] (“In FIG. 7D, the user completes the unlock action ... [such that] [t]he unlock image 702 and the channel 704 disappear

from the display and the virtual buttons 708 are at their final optical intensity levels, as illustrated by their solid outlines.”). At no point does an object associated with the first user-interface state (e.g., 702, 704, 706) decrease in optical intensity during the unlock action (FIGs. 7A-7C).

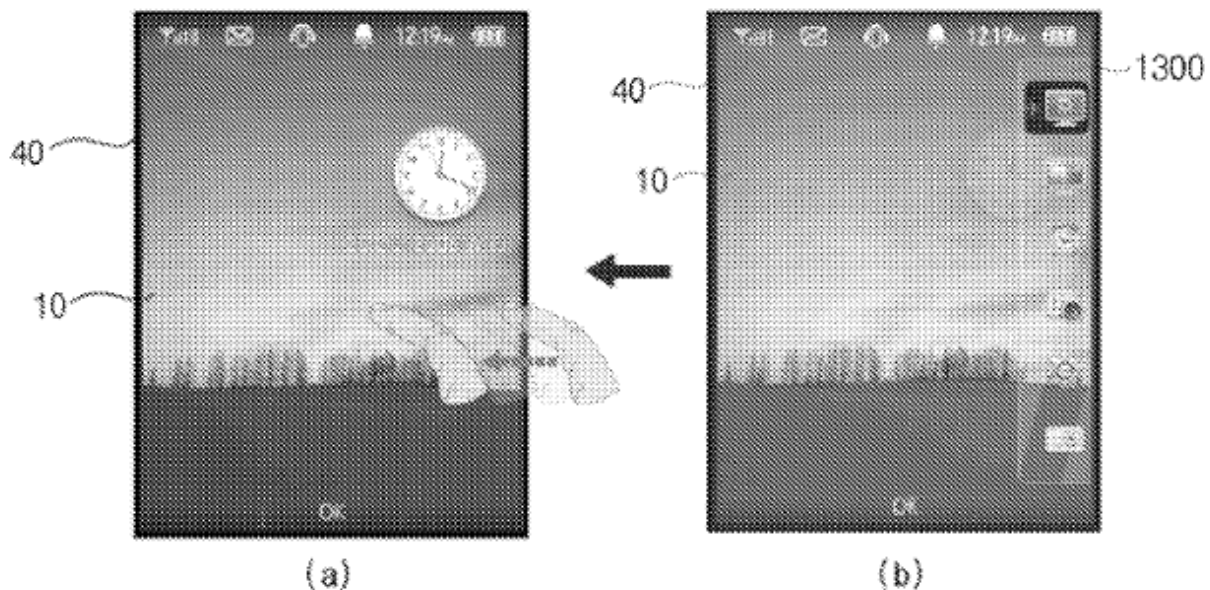


EX1005, FIG. 7A-7D.

104. In contrast, Dr. Terveen’s proposed combination of Ahn and Chaudhri relies on *reducing* the optical intensity of Ahn’s background (mapped to Chaudhri’s *first* user interface state). Specifically, Dr. Terveen identifies the standby screen of Ahn (FIG. 13A – below left) as corresponding to the first user-interface state and the

“state where a menu list 1300 is displayed” (FIG. 13B – below right) as the second user-interface state. EX1003, ¶76.

FIG. 13



EX2005, 71/75.

105. In Dr. Terveen’s proposed mapping to Ahn, the clock widget is displayed in (e.g., associated with) the *first* user-interface state. But Chaudhri-842 does not teach adjusting the optical intensity of an object associated with the *first* user-interface state. Thus, Dr. Terveen’s proposed modification to change the appearance of Ahn’s clock widget (of the *first* user-interface state) is inconsistent with Chaudhri-842’s teaching to “transition[] an optical intensity of one or more user interface objects associated with the *second user-interface state*.” EX1003, ¶82; As such, here too, Dr. Terveen’s proposed modification does the opposite of what

Chaudhri-842 teaches, and a POSITA would not have made such a conflicting modification—especially not without some articulable reason, which Dr. Terveen does not provide.

106. Ultimately, Dr. Terveen’s proposed modification would not have made sense in the context of Chaudhri-842’s teachings. As shown in Chaudhri-842’s FIGs. 7A-7D (above) a *selectable UI element* (708) from a second interface state increases in optical intensity to signify to the user that further completion of its gesture will allow the user to interact with that UI element. EX1005, ¶[0092]. In the example of Chaudhri-842’s FIGs. 7A-7D, the increasing optical intensity of the virtual buttons 708 helps the user understand that continuing to slide image 702 to the end of channel 704 will allow the user to interact with (e.g., select) one of buttons 708. *Id.*

107. In contrast, Dr. Terveen’s proposed modification would decrease the optical intensity of the *background* clock widget as the user drags their finger towards an undefined end, which would not offer any context to the user on what operation will occur or when. Thus, Dr. Terveen departs from Chaudhri-842’s teachings at least in that the proposed Ahn-Chaudhri-842 combination purports to make translucent a *background* object (clock widget) that is part of the *first* user interface state. It makes little sense to add translucency to a background, given the lack of anything beneath the background that might otherwise show through (Dr.

Terveen offers no other motivation for making any element translucent). This is evidenced by Ahn's disclosure in which only the menu is referred to as being translucent—not the background or anything else.

108. Moreover, once the touch and drag operation is complete and the menu subsequently deployed, the clock widget also cannot be interacted with (i.e., selected) in the second interface, meaning that the alleged *decreasing* of the optical intensity of the clock widget fails to serve the purpose set forth in Chaudhri-842 for *increasing* the optical intensity of Chaudhri-842's virtual buttons. In fact, a POSITA would understand that the clock widget of Ahn is not selectable when obstructed by menu. For example, if the clock widget was selectable when displayed at the same time of menu 420 it would cause significant UI issues as the user would not be able to reliably select between the clock widget and some icons (e.g., 420a, 420b). *See* EX1004, FIG. 4B.

109. Dr. Terveen fails to acknowledge these significant discrepancies, much less provide any evidence or explanation for why a POSITA would have done the opposite of what Chaudhri-842 discloses. That alone undermines the Dr. Terveen's proposed Ahn-Chaudhri-842 combination.

5. Dr. Terveen's Proposed Modification Would Not Achieve the Functionality Allegedly Motivating the Proposed Combination

110. Dr. Terveen relies on several of the benefits associated with Chaudhri-842's operation as motivations for a POSITA to make the proposed

modifications to Ahn. EX1003, ¶76. But, as explained above, Dr. Terveen’s proposed modifications are inconsistent with Chaudhri-842’s teachings and, ultimately, none of those purportedly motivating benefits actually carry over to Dr. Terveen’s proposed combination.

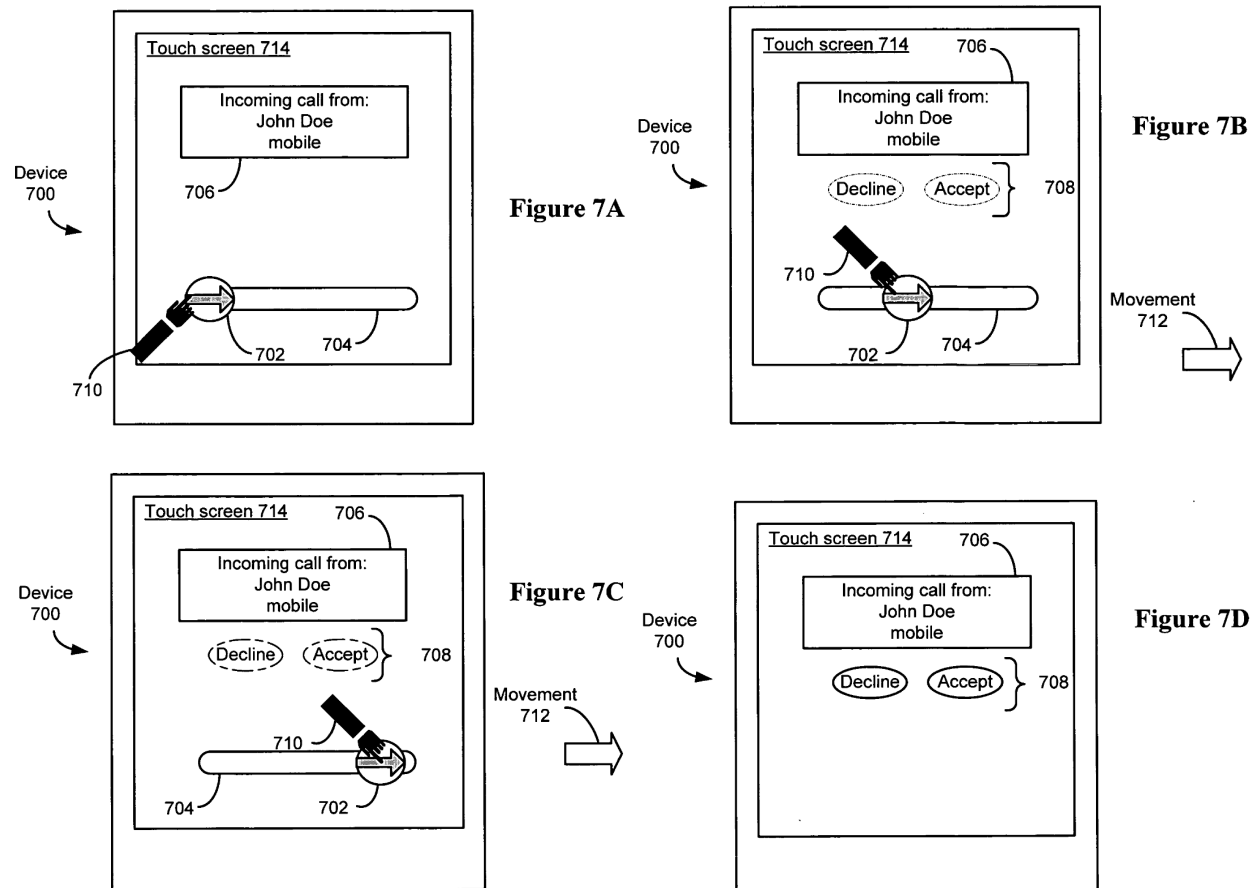
111. To illustrate, Dr. Terveen opines that it would have been obvious to implement the proposed modification—“blurring” of the clock widget during the touch and drag operation—“to provide *visual feedback* regarding the *progress* of a transition between user interface states.” EX1003, ¶72. According to Dr. Terveen, this would (1) “provid[e] a visual indication to the user of their progress in completing the gesture” and (2) “help differentiate between the” menu list and the clock widget. EX1003, ¶76. But a POSITA would not have expected either purported motivation to be present in Dr. Terveen’s proposed Ahn-Chaudhri-842 combination.

a) Dr. Terveen’s proposed modification does not provide visual feedback of the progress of a user’s gesture

112. The “blurring” of Ahn’s clock widget proposed by Dr. Terveen during Ahn’s “touch and drag operation” would not provide the user with any indication of progress of opening menu 420/1300.

113. Chaudhri-842 shows progress by an indicator 702 moving toward an “end” of a channel 704. EX1005, FIGs. 7A-7D; *see also id.*, [0071] (end of the channel 404 “serves as a predefined location to which the unlock image 402 is to be

dragged.”). Simultaneously, optical intensity of the virtual buttons 708 *increases* to “their final optical intensity levels, as illustrated by their ***solid outlines.***” EX1005, ¶[0093].



EX1005, FIGs. 7A-7D.

114. Thus, a user is given feedback of the progress of an “unlock action” by the distance the indicator 702 has moved relative to the total length of the channel 704, and by virtual buttons 708—associated with the second user-interface state—*increasing* in optical intensity to an opaque end point with “solid outlines.” *Id.*, [0090]-[0093]; *see also id.*, [0087] (“the indication of progress may be defined in

terms of the distance between the two edges because the distance remaining objectively measures how much further the user has to move her finger to complete the gesture”). Chaudhri-842’s visual feedback is thus provided only because the selectable virtual buttons 708 of the second user-interface state *increase* in optical intensity.

115. A POSITA would have understood that this visual feedback is absent from Dr. Terveen’s proposed modification to “blur” the clock widget of Ahn. *See* EX1003, ¶76. In Dr. Terveen’s proposed modification, the user cannot compare its gesture to some visual end point and, instead, would see only in abstract the proposed “blurring” of the clock widget. But that change in appearance of the clock widget offers no indication of progress, at least because its end state cannot be perceived or predicted based on that alleged change in appearance. Nor does it direct a user to a selectable object as in Chaudhri-842 because a user would not be able to interact with Ahn’s clock widget once the menu is displayed over the top of that clock widget. Nor is this end-state (display of the menu) otherwise related to the alteration of the clock widget in any manner that is perceivable by a user. For example, there is no frame of reference to discern any progress, relative to its required distance, of the gesture based on the changing appearance of the clock widget.

116. As such, the alleged “blurring” *during* the touch and drag operation would not provide any indication of (1) how much of the gesture is completed, or

(2) how much of the gesture is remaining. *Contra* EX1005, ¶¶[0088]-[0090] (“If the user input condition includes a predefined gesture, then the indication of progress of the gesture may be defined in terms of how much of the gesture is completed and ***how much of the gesture is remaining***”); EX1003, ¶75 (quoting same). In other words, a user cannot tell from the alleged “blurring” how much more “blurring” or swiping is needed before the menu will be completed.

117. Further, Dr. Terveen’s proposed modification to have menu 1300 slide out with graphical shadow 1800 “as part of the touch and drag gesture” (EX1003, ¶¶78-79) renders the “blurring” of the clock widget even less relevant. A user would know the gesture is completed when the menu 1300 stops advancing according to the user’s drag. There is no evidence or logical explanation to suggest that the “blurring” of the clock widget—to a state that is not completely transparent (EX1004, FIGs. 4B, 13B)—would suggest the gesture is complete or how much of that gesture remains until the menu is fully opened. Accordingly, a POSITA would have understood that Dr. Terveen’s purported modification to blur the clock widget would not provide any indication of a user’s progress in completing Ahn’s “touch and drag” gesture because, unlike Chaudhri-842, Dr. Terveen’s proposed UI shows no indication of progress toward a clearly visible end goal. And Dr. Terveen’s remaining modifications (to fade out the graphical shadow 1800 or slide in menu 1300) do not meet the claim language. *Supra*, §IX.A (Dr. Terveen only identifies

Ahn's clock widget as the claimed "at least one other object."). Thus, Dr. Terveen's purported 'progress' motivation would not have led a POSITA to make the proposed modification of Ahn.

b) Dr. Terveen's proposed modification does not help differentiate between objects

118. Dr. Terveen's assertion that "utilizing this progressive translucency or blurring effect would have helped differentiate between the user-interface object (e.g., the menu list 1300) and the background (e.g., the standby screen and/or clock widget)" is misguided and is not supported by evidence. EX1003, ¶76.

119. As explained above, the menu 1300 in Ahn is not selectable until the gesture is complete. Additionally, Dr. Terveen does not modify Ahn so that the clock widget is "blurred" more than already pictured in Ahn (e.g., transparency of the clock widget in Dr. Terveen's proposed modification is not reduced as compared to Ahn's FIG. 4B). *See* EX1003, ¶71-83. When displayed in a selectable state, the differentiation of the clock widget and the menu is identical in Ahn's disclosure and Dr. Terveen's proposed modification. Thus, this alleged 'differentiation' motivation offers no change from the 'differentiation' already provided by Ahn. That is, the alleged motivation provides no actual solution and is based on a problem that does not exist.

120. Further, a POSITA would understand that the cited portion of Chaudhri-842 teaches that differentiation between a user-interface element and a

background is a function of the total difference of optical intensity and not any *progressive change* in optical intensity. EX1005, ¶[0085] (defining “*the* optical intensity”—not a progressive change in optical intensity—as “the visual differentiation between the user-interface object and the background”); EX1003, ¶74 (citing EX1005, ¶[0085]). Thus, it is the difference in opacity between the UI element and the background that results in the visual differentiation of Chaudhri-842. EX1005, ¶[0085]. Accordingly, a POSITA reading Chaudhri-842 would have understood that progressively changing the appearance of Ahn’s background would not improve visual differentiation of Ahn’s menu 1300. Thus, Dr. Terveen’s purported ‘differentiation’ motivation would not have led a POSITA to make the asserted modification of Ahn.

c) Dr. Terveen’s generic motivations also fail

121. Dr. Terveen’s mapping of Limitation [1.d] also references Section VIII.C of his own declaration as support for its assertion that “a POSITA would have found it obvious to apply Chaudhri-842’s user interface state transition techniques—particularly the use of transitioning optical intensity to provide visual feedback regarding the progress of a transition between user interface states as a result of detecting a user’s input of a predefined gesture—in the context of Ahn’s mobile communication terminal 100.” EX1003, ¶72. Foundationally, this sentence is so broad as to be meaningless and, as explained above, Dr. Terveen’s proposed

modification does not “provide visual feedback regarding the progress of a transition between user interface states.” *Supra*, §IX.A.5.a). Dr. Terveen fails to explain why Chaudhri-842’s teachings would be applied in a way that would meet the claims. Nevertheless, the four motivations alleged in that referenced section also fail to lead a POSITA to Dr. Terveen’s proposed combination. *See* EX1003, ¶¶50-55.

122. First, Dr. Terveen asserts generally that “Chaudhri-842 provides one example of implementation details that are lacking for a feature otherwise disclosed in Ahn.” EX1003, ¶50. However, for the reasons explained above in Sections IX.A.4.b), Chaudhri-842 (1) discloses varying the optical intensity of foreground UI elements (not the background), (2) discloses increasing the optical intensity of UI elements associated with a second user-interface state (not decreasing optical intensity of a background associated with first user-interface state), and (3) these foreground UI objects associated with a second user-interface state can be interacted with once the increase in their optical intensity has been completed. As such, Dr. Terveen’s proposed modifications to Ahn do not implement the alleged “implementation details” of Chaudhri-842’s. *Id.*

123. Second, Dr. Terveen asserts generally that “Chaudhri-842 teaches that the ‘change in optical intensity of the user-interface objects provides a user with sensory feedback of the progress in transitioning between user interface states.’”

EX1003, ¶52. But this is the same motivation parroted specifically for Limitation [1.d.i], and therefore fails for the same reasons explained in Section IX.A.5.a) above.

124. Third, Dr. Terveen asserts generally that “a POSITA would have known that transparency, translucency/blurring, and other ‘focus and context’ techniques may be utilized in user interfaces to help draw a user’s attention to the most relevant part of a user interface.” EX1003, ¶54. This too is the same “visual differentiation” motivation parroted specifically for Limitation [1.d.i], and therefore fails for the same reasons explained in Sections IX.A.5.a)-b).

125. Fourth, Dr. Terveen asserts generally that “POSITA would have known that varying optical intensity for a user interface object (e.g., the blurring of Ahn’s clock widget) as a function of the magnitude of the user’s touch and drag gesture would have been one of a limited number of ways of implementing the blurring effect that’s shown in FIGS. 4B, 10, and 13B of Ahn.” EX1003, ¶55. Dr. Terveen continues that “[e]ither a POSITA could have implemented the blurring effect in a binary fashion (e.g., toggling instantaneously between unblurred and blurred) or progressively varied the blur as a function of the user’s gesture, as taught by Chaudhri-842.” EX1003, ¶55. Here, Dr. Terveen abandons his first three motivations and, as a last-ditch fallback, constructs a false binary choice of its own by framing the decision as simply either/or. Not only is this motivation divorced from the teachings of Ahn or Chaudhri-842, but it is also belied by Dr. Terveen’s

own opinions based on the various reasons and considerations driving a POSITA's decision of whether, when, and how to alter the appearance of Ahn's background. *See, e.g.*, EX1003, ¶¶50-55.

126. As explained in Section IX.A.3 above, Ahn's widget is not "blurred" until after the "touch and drag operation" is complete. Weighing against doing so is the fact that varying the optical intensity of the background is not without cost. In particular, *progressively* varying the optical intensity of the background in a touch input device 10 like Ahn's would require code that would eat into the device's limited storage, as well as the use of limited processing and memory bandwidth, and limited battery power, to execute the function of varying that optical intensity. In Dr. Terveen's proposed combination, progressively varying the optical intensity of the background while the menu 1300 is not selectable would serve no purpose. As such, even assuming Dr. Terveen's false binary, a POSITA would have chosen not to expend resources to make that modification to Ahn.

127. For the reasons explained above, Dr. Terveen's modification would not achieve any of the purported benefits alleged to motivate Dr. Terveen's modifications of Ahn.

X. CONCLUSION

128. All statements made herein of my own knowledge are true, all statements made herein on information and belief are believed to be true, and these statements were made with the knowledge that willful false statement and the like are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

Respectfully submitted,

Dated: December 18, 2024



Rajeev Surati, Ph.D.