

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of: Michael S. Smith
U.S. Patent No.: 10,936,114 Attorney Docket No. 50095-0171IP1
Issue Date: March 2, 2021
Appl. Serial No.: 16/558,026
Filing Date: August 30, 2019
Title: GESTURE-EQUIPPED TOUCH SCREEN SYSTEM, METHOD,
AND COMPUTER PROGRAM PRODUCT

DECLARATION OF DR. ANDREW COCKBURN

I currently hold the opinions set expressed in this declaration. But my analysis may continue, and I may acquire additional information and/or attain supplemental insights that may result in added observations.

I hereby declare that all statements made of my own knowledge are true and that all statements made on information and belief are believed to be true. I further declare that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of the Title 18 of the United States Code.

Dated: 26th June, 2024

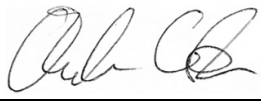
By: 
Andrew Cockburn

TABLE OF CONTENTS

I.	QUALIFICATIONS AND BACKGROUND INFORMATION.....	iv
II.	LEGAL PRINCIPLES.....	2
	A. Anticipation.....	2
	B. Obviousness	3
III.	OVERVIEW OF CONCLUSIONS FORMED	4
IV.	LEVEL OF ORDINARY SKILL IN THE ART	5
V.	INTERPRETATIONS OF THE '114 PATENT CLAIMS AT ISSUE.....	6
VI.	OVERVIEW OF THE '114 PATENT	6
	A. The '114 Patent	6
	B. Prosecution History.....	8
VII.	PRIOR ART ANALYSIS.....	9
	A. Ground 1A: Claims 1, 27, and 32 are anticipated and/or rendered obvious over Kim.....	9
	1. Kim	9
	2. Claim 1	18
	3. Claim 27	37
	4. Claim 27	44
	5. Claim 32	50
	B. Ground 1B: Claims 34, 59, and 64 are obvious over Kim in view of Lang 55	55
	1. Lang.....	55
	2. The Kim-Lang Combination	56
	3. Claim 34	57
	4. Claim 59	59
	5. Claim 64	60
	C. Ground 1C: Claim 30 is obvious over Kim in view of Burrough	61
	1. Burrough.....	61
	2. The Kim-Burrough Combination	63
	3. Claim 30	67
	D. Ground 1D: Claim 62 is obvious over Kim in view of Lang and Burrough	72
	1. The Kim-Lang-Burrough Combination	72
	2. Claim 62	73
	E. Ground 2A: Claims 1, 27, and 32 are obvious over Kim in view of Bowens.....	74
	1. Bowens	74

2.	The Kim-Bowens Combination	75
3.	Claim 1	90
4.	Claim 27	97
5.	Claim 32	104
F.	Ground 2B: Claims 34, 59, and 64 are obvious over Kim in view of Bowens and Lang.....	109
1.	The Kim-Bowens-Lang Combination.....	109
2.	Claim 34	109
3.	Claim 59	111
4.	Claim 64	112
G.	Ground 2C: Claim 30 is obvious over Kim in view of Bowens and Burrough	113
1.	The Kim-Bowens-Burrough Combination.....	113
2.	Claim 30	116
H.	Ground 2D: Claim 62 is obvious over Kim in view of Bowens, Lang, and Burrough	119
1.	The Kim-Bowens-Lang-Burrough Combination.....	119
1.	Claim 62	120

I. QUALIFICATIONS AND BACKGROUND INFORMATION

1. My education and experience are described more fully in the attached curriculum vitae (Appendix A). I am a Professor at the Department of Computer Science and Software Engineering at the University of Canterbury, New Zealand. I also currently head the HCI (Human-Computer Interaction) and Multi-Media research group at the University of Canterbury.

2. In 1988, I was awarded a Bachelor of Science with Honors in Computer Science from the University of York, England.

3. In 1993, I was awarded a Ph.D. from the University of Stirling, Scotland. My thesis was on “Computer Supported Cooperative Work” which relates to forms of group interaction supported on computers.

4. In 1993, I joined the University of Canterbury as a Lecturer in the Department of Computer Science (now the Department of Computer Science and Software Engineering). I was subsequently promoted to a Senior Lecturer, and then an Associate Professor, before my appointment as a Professor in 2010. I currently hold the title of Professor.

5. I have over 30 years of experience in the area of HCI. The field of HCI is generally concerned with ways of understanding and improving the interaction between humans and computers, with a view to understanding, evaluating, designing, and building new styles of interactions that improve on one

or more of the end goals of making computers faster to learn, more intuitive, more efficient to use, and more subjectively satisfying.

6. Throughout my career, I have published the results of many research projects that have involved building new user interfaces or reviewing existing user interfaces for performing a particular task, and evaluating their effectiveness. This includes publications relating to:

- (a) investigating user experiences with web navigation interfaces, for example when using the “back” button on web browsers;
 - (b) reviewing and improving various interface schemes for traversing through documents and lists in desktop and mobile computer applications, including zooming, scrolling, and other techniques;
 - (c) analyzing new interfaces for text entry on mobile and touch-sensitive devices;
 - (d) addressing the problems arising from the small form factor of mobile devices with touch-sensitive displays, including wearable technologies;
 - (e) examining the influence of haptic feedback on user performance with mouse and touchscreen input devices;
 - (f) evaluating the importance of spatially stable displays in user interfaces;
- and

(g) designing, analyzing and evaluating gesture-based input techniques for various user activities, including music data entry, web navigation, text entry, and scrolling.

7. I also have extensive experience in designing and building new user interfaces and reviewing existing user interfaces. This includes a number of projects regarding the design, development and evaluation of user interfaces that I have undertaken with companies in the computing and HCI industry, such as:

(a) working with Airbus SAS in 2016 on methods to assist pilots' gesture-based interaction with touchscreens in turbulent environments;

(b) working with Hewlett-Packard Research Labs from 2010-2012 on the design and evaluation of gesture-based pointing techniques for remote displays, such as interactive TVs;

(c) a number of projects from 2006-2010 working with Logitech on the design, development, evaluation and improvement of user interfaces for next generation mice, including in relation to scrolling and window management tasks;

(d) working with IBM Almaden Research in 2006 on the design, development and evaluation of user interfaces for gesture-based text entry on mobile devices;

- (e) working with Digit Wireless in 2002 on the evaluation of user performance for user interfaces for digital text entry on mobile devices; and
- (f) working with Microsoft Research in 1999 on the development, evaluation and improvement of user interfaces for web browsing, in particular the mechanisms for revisiting pages (such as through the “back” button or bookmarks).

8. At the University of Canterbury, I currently teach a course on HCI for final year undergraduate students and a course on research methods in HCI for postgraduate students.

9. In the past, I have also taught introductory computer programming courses for first year students across disciplines, which included the implementation of graphical user interfaces.

10. I also manage an active research lab with a number of Ph.D. students. I have previously supervised fourteen students to successful completion of their Ph.D.s in the field of HCI.

11. The Association for Computing Machinery (ACM) is the world’s largest scientific and educational computing society. In 2015, I was elected to the ACM Special Interest Group on Computer-Human Interaction (CHI) Academy, which honors the principal leaders of the field of HCI.

12. I also annually attend conferences relating to the field of HCI and regularly read journals that cover research in the field of HCI.

13. To keep abreast with industry trends, I regularly read broader journals, magazines and websites related to computer science and technology, including Communications of the ACM, ACM TechNews, BBC Technology, and New York Times Technology.

14. A detailed list of my other professional activities, memberships, and speaking engagements is included in my CV.

15. I have been retained on behalf of Apple Inc. to offer technical opinions relating to U.S. Patent No. 10,936,114 (“the ’114 Patent”) and prior art references relating to its subject matter. I have reviewed the ’114 Patent, and relevant excerpts of the prosecution history of the ’114 Patent. I have also reviewed the following references:

- APPLE-1001 U.S. Patent No. 10,936,114 (“the ’114 Patent”)
- APPLE-1002 U.S. Patent No. 10,936,114 File History
- APPLE-1004 U.S. Patent Publication No. 2010/0302281 to Kim (“Kim”)
- APPLE-1005 U.S. Patent Publication No. 2009/0273571 to Bowens (“Bowens”)
- APPLE-1006 U.S. Patent Publication No. 2006/0139320 to Lang (“Lang”)

- APPLE-1007 U.S. Patent Publication No. 2010/0156818 to Burrough et al. (Burrough)
- APPLE-1008 U.S. Patent Publication No. 2011/0310043 to Lacroix et al. (Lacroix)
- APPLE-1009 Simon, Windows CE 2 for Dummies (1997)
- APPLE-1010 U.S. Patent Publication No. 2011/0221666 to Newton et al. (“Newton”)
- APPLE-1011 U.S. Patent Publication No. 2015/0324116 to Marsden et al. (“Marsden”)
- APPLE-1012 U.S. Patent Publication No. 2018/0059928 to Westerman et al. (“Westerman”)
- APPLE-1013 U.S. Patent Publication No. 2013/0212541 to Dolenc et al. (“Dolenc”)
- APPLE-1014 U.S. Patent Publication No. 2005/0168488 to Montague (“Montague”)
- APPLE-1015 U.S. Patent No. 8,954,887 to Tseng et al. (“Tseng”)
- APPLE-1016 Smith’s Infringement Contentions
- APPLE-1017 U.S. Patent Publication No. 2007/0024593 to Schroeder (“Schroeder”)
- APPLE-1018 U.S. Patent Publication No. 2009/0167508 to Fadell et al. (“Fadell”)

16. I have also reviewed various supporting references and other documentation as further noted in my opinions below.

17. Counsel has informed me that I should consider these materials through the lens of one of ordinary skill in the art related to the '114 Patent at the time of the earliest possible priority date of the '114 Patent, and I have done so during my review of these materials. The '114 Patent claims the benefit of priority to a number of provisional applications, the earliest of which was filed on August 5, 2011 ("the Critical Date"). Counsel has informed me that the Critical Date represents the earliest possible priority date to which the challenged claims of '114 Patent are entitled, and I have therefore used that Critical Date in my analysis below.

18. I have no financial interest in the parties or in the outcome of this proceeding. I am being compensated for my work as an expert on an hourly basis. My compensation is not dependent on the outcome of these proceedings or the content of my opinions.

19. In writing this declaration, I have considered the following: my own knowledge and experience, including my work experience in the field of HCI; my experience in teaching those subjects; and my experience in working with others

involved in those fields. In addition, I have analyzed various publications and materials, in addition to other materials I cite in my declaration.

20. My opinions, as explained below, are based on my education, experience, and expertise in the fields relating to the '114 Patent. Unless otherwise stated, my testimony below refers to the knowledge of one of ordinary skill in the art as of the Critical Date, or before. Any figures that appear within this document have been prepared with the assistance of Counsel and reflect my understanding of the '114 Patent and the prior art discussed below.

II. LEGAL PRINCIPLES

A. Anticipation

21. I have been informed that a patent claim is invalid as anticipated under 35 U.S.C. § 102 if each and every element of a claim, as properly construed, is found either explicitly or inherently in a single prior art reference. Under the principles of inherency, if the prior art necessarily functions in accordance with, or includes the claimed limitations, it anticipates.

22. I have been informed that a claim is invalid under 35 U.S.C. § 102(a) if the claimed invention was known or used by others in the U.S., or was patented or published anywhere, before the applicant's invention. I further have been informed that a claim is invalid under 35 U.S.C. § 102(b) if the invention was patented or published anywhere, or was in public use, on sale, or offered for sale in

this country, more than one year prior to the filing date of the patent application (critical date). And a claim is invalid, as I have been informed, under 35 U.S.C. § 102(e), if an invention described by that claim was described in a U.S. patent granted on an application for a patent by another that was filed in the U.S. before the date of invention for such a claim.

B. Obviousness

23. I have been informed that a patent claim is invalid as “obvious” under 35 U.S.C. § 103 in light of one or more prior art references if it would have been obvious to a POSITA, taking into account (1) the scope and content of the prior art, (2) the differences between the prior art and the claims, (3) the level of ordinary skill in the art, and (4) any so called “secondary considerations” of non-obviousness, which include: (i) “long felt need” for the claimed invention, (ii) commercial success attributable to the claimed invention, (iii) unexpected results of the claimed invention, and (iv) “copying” of the claimed invention by others. For purposes of my analysis above and because I know of no indication from the patent owner or others to the contrary, I have applied a date of August 5, 2011, as the date of invention in my obviousness analyses, although in many cases the same analysis would hold true even at an earlier time than August 5, 2011.

24. I have been informed that a claim can be obvious in light of a single prior art reference or multiple prior art references. To be obvious in light of a

single prior art reference or multiple prior art references, there must be a reason to modify the single prior art reference, or combine two or more references, in order to achieve the claimed invention. This reason may come from a teaching, suggestion, or motivation to combine, or may come from the reference or references themselves, the knowledge or “common sense” of one skilled in the art, or from the nature of the problem to be solved, and may be explicit or implicit from the prior art as a whole. I have been informed that the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. I also understand it is improper to rely on hindsight in making the obviousness determination.

III. OVERVIEW OF CONCLUSIONS FORMED

25. This declaration explains the conclusions that I have formed based on my analysis. To summarize those conclusions, based upon my knowledge and experience and my review of the prior art publications listed above, I believe that:

- Claims 1, 27, and 32 are anticipated and/or obvious over Kim
(Ground 1A)
- Claims 34, 59, and 64 are obvious over Kim in view of Lang (Ground 1B)
- Claim 30 is obvious over Kim in view of Burrough (Ground 1C)

- Claim 62 is obvious over Kim in view of Lang and Burrough (Ground 1D)
- Claims 1, 27, and 32 are obvious over Kim in view of Bowens (Ground 2A)
- Claim 34, 59, and 64 are obvious over Kim in view of Bowens and Lang (Ground 2B)
- Claim 30 is obvious over Kim in view of Bowens and Burrough (Ground 2C)
- Claim 62 is obvious over Kim in view of Bowens, Lang, and Burrough (Ground 2D)

IV. LEVEL OF ORDINARY SKILL IN THE ART

26. In my opinion, a person of ordinary skill in the art (“POSITA”) relating to, and at the time of, the purported invention of the ’114 Patent would have had at least a bachelor’s degree in computer science, computer engineering, electrical engineering, human-computer interaction, or a related field, and would have had at least two years of relevant work experience in the design and development of graphical user interfaces (GUIs), human-computer interfaces, or equivalents thereof. Less work experience may be compensated by a higher level of education and vice versa.

27. Based on my experiences, I have a good understanding of the capabilities of one of ordinary skill. Indeed, I have taught, participated in organizations, and worked closely with many such persons over the course of my career. Based on my knowledge, skill, and experience, I have an understanding of the capabilities of one of ordinary skill. Furthermore, I possess those capabilities myself.

V. INTERPRETATIONS OF THE '114 PATENT CLAIMS AT ISSUE

28. I have been informed by Counsel and understand that the best indicator of claim meaning is its usage in the context of the patent specification as understood by one of ordinary skill. I further understand that the words of the claims should be given their plain meaning unless that meaning is inconsistent with the patent specification or the patent's history of examination before the Patent Office. Counsel has also informed me, and I understand that, the words of the claims should be interpreted as they would have been interpreted by one of ordinary skill at the time of the invention was made (not today). I have been informed by Counsel that I should use August 5, 2011 as the point in time for claim interpretation purposes.

VI. OVERVIEW OF THE '114 PATENT

A. The '114 Patent

29. In a brief Background section, the '114 Patent states that “the present invention generally relate to touch screens.” APPLE-1001, 1:37-38.

30. In particular, the '755 purports to provide an apparatus including: “at least one non-transitory memory storing instructions and a plurality of applications; 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 the instructions to: display, via the touch screen, an object and at least a portion of an interface; detect a gesture on the touch screen on the object; when a magnitude of a duration of the gesture being detected on the touch screen on the object is less than a first magnitude threshold, perform an operation; when the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than the first magnitude threshold, perform a scaling in connection with at least the object, where the scaling includes a reduction in a size such that the size is inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the object; and when the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than a second magnitude threshold, perform another operation.” APPLE-1001, 1:42-61.

31. Claim 27 of the '114 Patent, for example, recites “[a]n apparatus” comprising, among other components, “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 the instructions to cause the apparatus to:

- display an object on at least a portion of an interface;
- detect a gesture via the touch screen on the object;
- perform a first function in a first touch state;
- perform a second function in a second touch state; and
- during detection of at least a portion of the gesture and based on a change in a duration of the gesture being detected via the touch screen on the object, perform a scale operation in connection with the at least portion of the interface, where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected via the touch screen on the object.

B. Prosecution History

32. Application Serial No. 16/558,026, which later issued as the '114 Patent, was filed August 30, 2019, with 28 claims, which were soon after canceled with claims 29-58 added. APPLE-1002, 911-924, 741-764.

33. During prosecution, the examiner issued no Office action, and thus, no prior art rejections. *See, generally, id.* Instead, Applicant offered a “Proposed Amendment,” which the parties discussed in an interview, and the examiner

subsequently allowed the claims without any prior art rejection. *Id.*, 305-358, 134, 74-133.

34. Notably, however, despite the hundreds of pages of references provided to the Office, the examiner did not issue a single prior art rejection before allowing the claims. This petition addresses claim features distinguished during prosecution in a fresh way using new combinations of prior art, none of which were previously considered.

VII. PRIOR ART ANALYSIS

A. Ground 1A: Claims 1, 27, and 32 are anticipated and/or rendered obvious over Kim

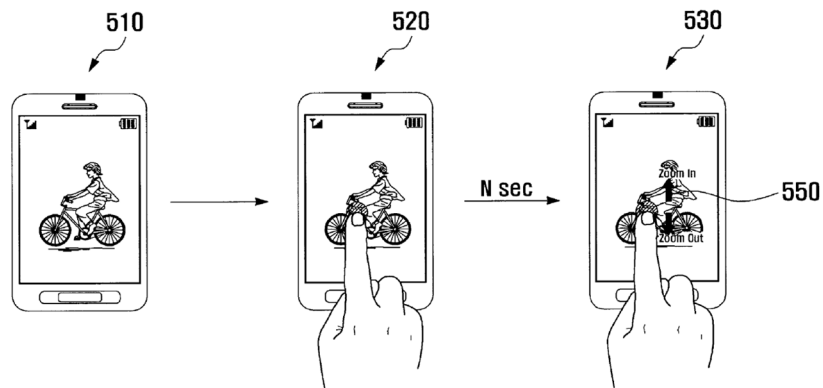
1. Kim¹

35. Kim describes a mobile device that “has a touch sensitive surface such as a touch screen” that provides “a touch-based zooming control method” for “portable and handheld devices.” APPLE-1004, Abstract, [0118]. At a high level, Kim explains that, “while performing a selected mode” (i.e., “a particular mode that allows displaying some kind of display target data outputted while a specific application selected at a user’s request runs in the mobile device”), “the mobile

¹ General descriptions provided for this and other references and combinations thereof are incorporated into each subsection and mapping of the claims that includes citations to these references.

device detects the occurrence of an interrupt,” which is “a signal or instructions, inputted through a touch screen or a special function key by a user, to stop the execution of a running application.” *Id.*, [0048]. “After detecting an interrupt, the mobile device determines whether an interrupt is a request for activating the zoom mode,” including, for example, determining “whether an interrupt is a long press input.” *Id.*, [0049]. Kim further explains that the interrupt may be, for example, “a request for the zoom mode” or a request for “some action in connection with the interrupt,” such as “a selection input for a specific icon” or “a request for a specific menu.” *Id.*, [0049]-[0051]. In any case, when Kim’s device “detect[s] a user gesture” by “receiving an input event from a user through the touch screen, in response, the mobile device determines the type of an input event,” including, for example, “whether an input event is one of a move event ... , a hold event ..., and a double tap event.” *Id.*, [0053]. If the detected input is none of them, “the mobile device disregards the received input event.” *Id.*

36. In Kim, while “target data is displayed, the mobile device activates a zoom mode in response to a given gesture or input event,” and “executes a zooming function such as a zoom-in and a zoom-out in response to the received gesture input.” APPLE-1004, Abstract. Figure 5 “illustrates an example of a process of activating a zoom mode of a mobile device.” *Id.*, [0083]. Kim explains that the central point of the zoom can be determined by a tap. *Id.*, [0066]-[0067].

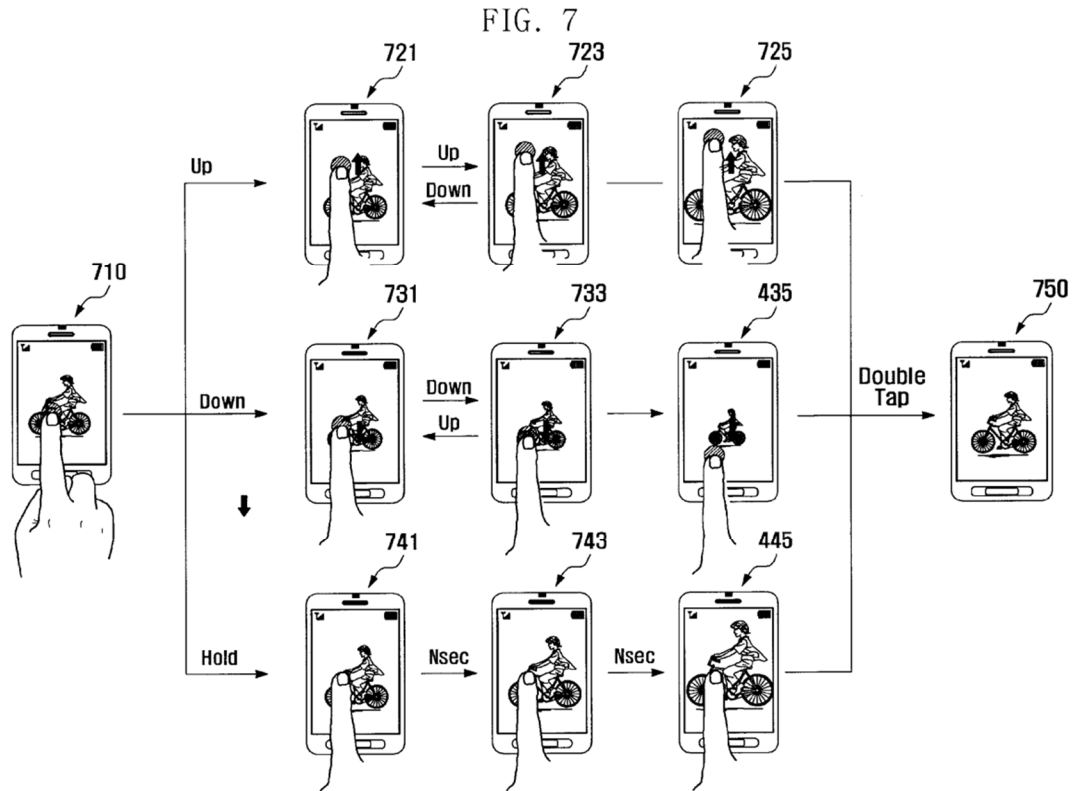


APPLE-1004, FIG. 5.

37. Kim explains that “a zoom mode can be activated when a user invokes a *long press* at a certain point (x, y) on a touch screen” and “can perform a zooming action in the following two ways.” APPLE-1004, [0139] (emphasis added). First, a user can move his finger up or down “from a pressed point (x, y) in order to perform a zoom-in or a zoom-out around that point (x, y).” *Id.*, [0140]. Or second, “a user can hold one of his or her finger ... without moving at a pressed point (x, y).” *Id.*

38. In more detail, Figure 7 “illustrates a process of executing the zooming function depending on a touch-based user gesture in a mobile device.” APPLE-1004, [0094]. As illustrated in Figure 7, “the mobile device can activate the zoom mode in response to a specific user gesture as discussed earlier in FIG. 5,” and after “the zoom mode is activated, a user can manipulate the mobile device

to perform a continuous zooming function (namely, a zoom-in or a zoom-out) through an appropriate gesture such as a move event or a hold event.” *Id.*, [0095].



APPLE-1004, FIG. 7.

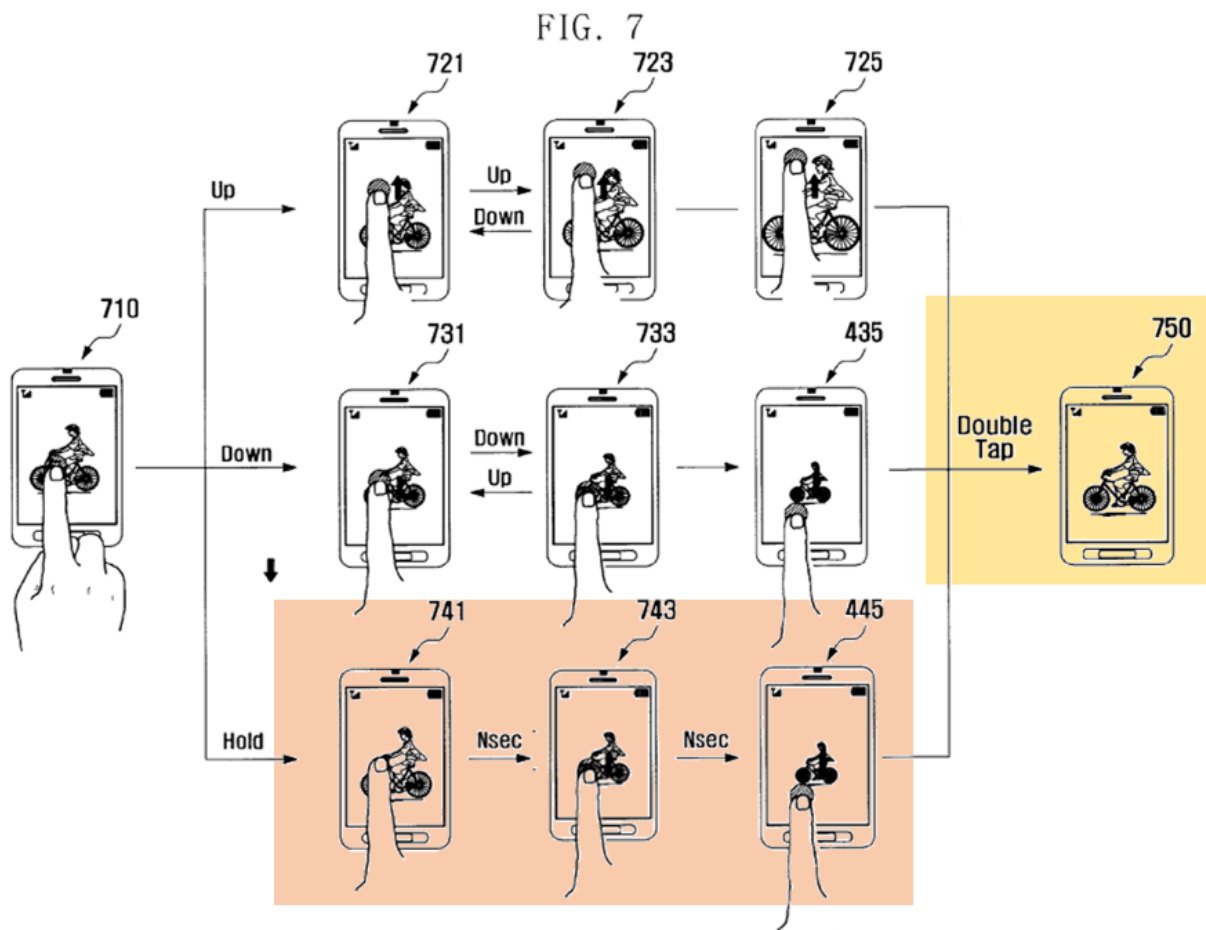
39. In the top two examples above, a user can “continuously invoke” either a move up or move down event at a selected point to cause a continuous zoom-in or zoom-out function, respectively. APPLE-1004, [0096]-[0103]. In another example, shown in the bottom path of Figure 7, “reference numbers 741, 743 and 745 show a continuous zoom-in action depending on a hold event.” APPLE-1004, [0104]. Specifically, “a user can invoke the hold event at a selected point through a touch input such as tap and hold,” and “the mobile device executes

a continuous zoom-in function around the selected point at given periodic intervals while the hold event continues.” *Id.* That is, as shown in Figure 7, the user touches the displayed person on a bike, or just the bike (“selected point”), and the device executes a zoom-in function around that point on the bike, as evidenced by the bike (and its surrounding data) being displayed at a larger size. *Id.* In this way, Kim teaches or at least renders obvious that the user input is received on the object that is later scaled up (zoomed-in) or scaled down (zoomed-out). *Id.* Put another way, Kim teaches or renders obvious that the user touches the portion of the screen from which the zooming function is centered. *Id.* In each of the figures showing a zooming action, the person on the bike is clearly modified by a zooming action, so a POSITA would have understood that the user gesture was received on the displayed person on a bike. *See, e.g., id.*, FIGS. 5, 7. Additionally, Figures 5, 6, and 7 all clearly show a user input on the person on the bike on the touch screen.

40. Moreover, Kim explains that in “a zoom-in process, the mobile device continuously offers a real-time, scaled-up bitmap image on the screen in response to the duration of the hold event,” and when “the hold event is released from the screen, the mobile device displays an enlarge display target data around the selected point on the touch screen, depending on a zoom-in ratio of a bitmap image.” *Id.*, [0105]. Kim further explains that “the hold event is assigned to a

continuous zoom-in function is an exemplary case only,” and “the hold event may be alternatively assigned to a continuous zoom-out function.” *Id.*, [0107].

41. Since “the hold event may be alternatively assigned to a continuous zoom-out function,” Kim teaches that bottom path of Figure 7 can be zoom-out instead of zoom-in, as illustrated in Figure 7 (modified), below (highlighted orange). APPLE-1004, [0107].



APPLE-1004, FIG. 7 (modified)

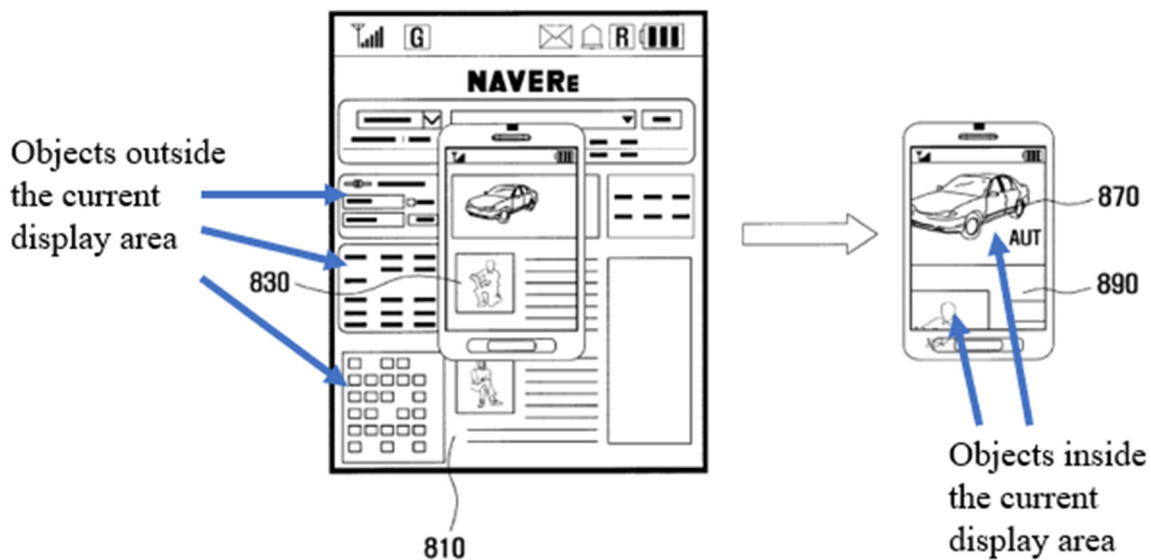
42. In addition to using a hold event, Kim further teaches that other gestures, such as press and move, can be used to control zooming. APPLE-1004,

[0094]-[0103]. Kim also teaches that, after a zoom action, “a user can invoke a double tap event on the screen,” which causes “the mobile device [to] display[] target data with a given basic magnification value, as indicated by reference number 750” in Figure 7 (modified) above (highlighted yellow). APPLE-1004, [0099], [0103], [0106], FIG. 7.

43. While Figure 7 shows only a rider on a bicycle,² Figure 8 clearly shows that scaling can be performed on objects other than the object with respect to which the user gesture is directed or detected. APPLE-1004, FIG. 8. Kim’s device, “controls a zooming action ... in response to a user gesture” where the displayed data is a portion of the web page that fits on the device screen’s interface. APPLE-1004, [0111]. While Figure 8 does not explicitly indicate where the user gesture was received, a POSITA would have understood that the gesture was received on a single object (as opposed to on the entire touch screen), and most likely, it was received on or near the car in view of the resulting enlarged display of the content 870. That is, Kim explains that the “zooming function” occurs “around a specific point on the target data,” so, given that the zoomed-in

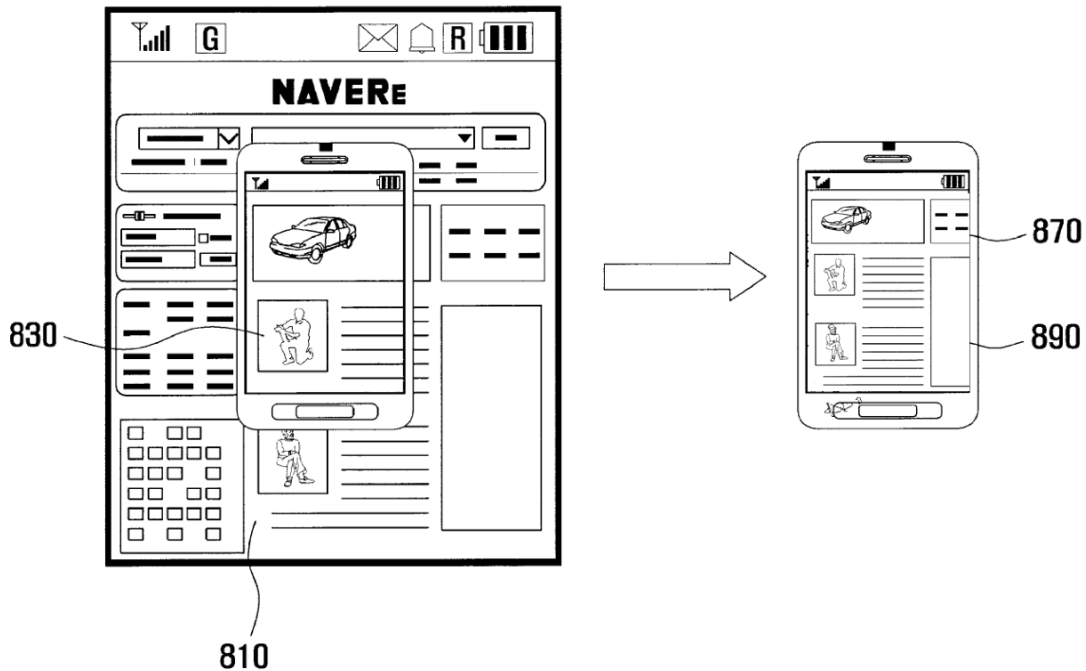
² Arguably, the rider on the bicycle shown in Figure 7 could be considered at least two separate objects (e.g., a rider and a bicycle), where the gesture is detected on the bicycle but not the rider. *See* APPLE-1004, FIG. 7 (741).

image is shown around the car, a POSITA would have understood that the user gesture was received on or near the car. Moreover, in response to the user gesture, the zoom function affects not just the car but also the content surrounding the car (e.g., the surrounding text and image). APPLE-1004, FIG. 8. Kim explains “a zooming action associated with web display data” where the displayed data is a portion of the web page that fits on the device screen’s interface. APPLE-1004, [0113]. Relatedly, in Figure 8, a viewport “contains a particular link assigned to certain text ‘AUTO’” and the “web display data is enlarged several times.” *Id.*, [0114]. Kim’s device performs “a continuous zooming action by using the bitmap image.” *Id.*, [0115].



APPLE-1004, FIG. 8 (annotated).

44. Per Kim's teachings, the zooming action shown in Figure 8 can also be a zoom-out action, as illustrated in modified FIG. 8 below. APPLE-1004, [0107], [0113].



APPLE-1004, FIG. 8 (modified)

45. In Kim, the “continuous zoom-in action depend[s] on a hold event” such that “the mobile device executes a continuous zoom-in function around the selected point at given periodic intervals while the hold event continues.” APPLE-1004, [0104]. In one example, “if a periodic interval is set for N seconds, the mobile device can perform a zooming control at intervals of N seconds while the hold event continues.” *Id.*, [0104]. In at least this way, Kim discloses performing another zoom operation or function based on duration of the detected gesture

magnitude. *Id.* Moreover, a POSITA would have understood from Kim's disclosure that, after the periodic interval (N seconds), the zooming would cease.

2. Claim 1

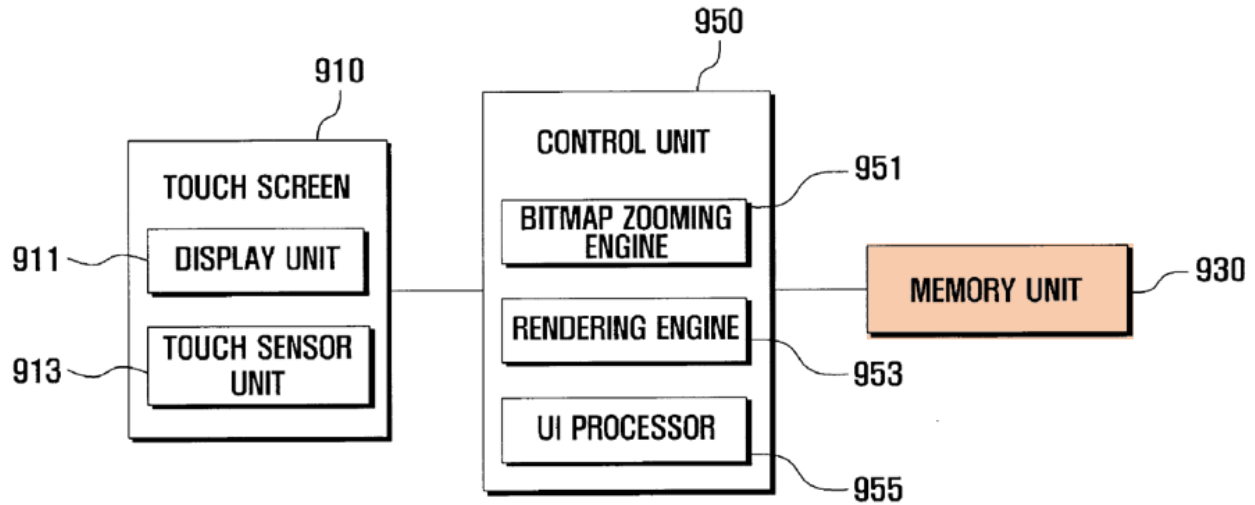
[1.pre] An apparatus, comprising:

46. To the extent the preamble is limiting, Kim teaches or renders [1.pre] obvious. Kim discloses a mobile device with "a touch sensitive surface such as a touch screen" that provides "a touch-based zooming control method." APPLE-1004, Abstract. The mobile device "includes a plurality of electronic devices having a touch sensitive surface such as one of a touch screen and a touch pad" and can be any other kinds of portable and handheld devices." *Id.*, [0118]. An exemplary mobile device is shown below in an excerpt from Kim's Figure 5.



[1.a] at least one non-transitory memory storing instructions and a plurality of applications;

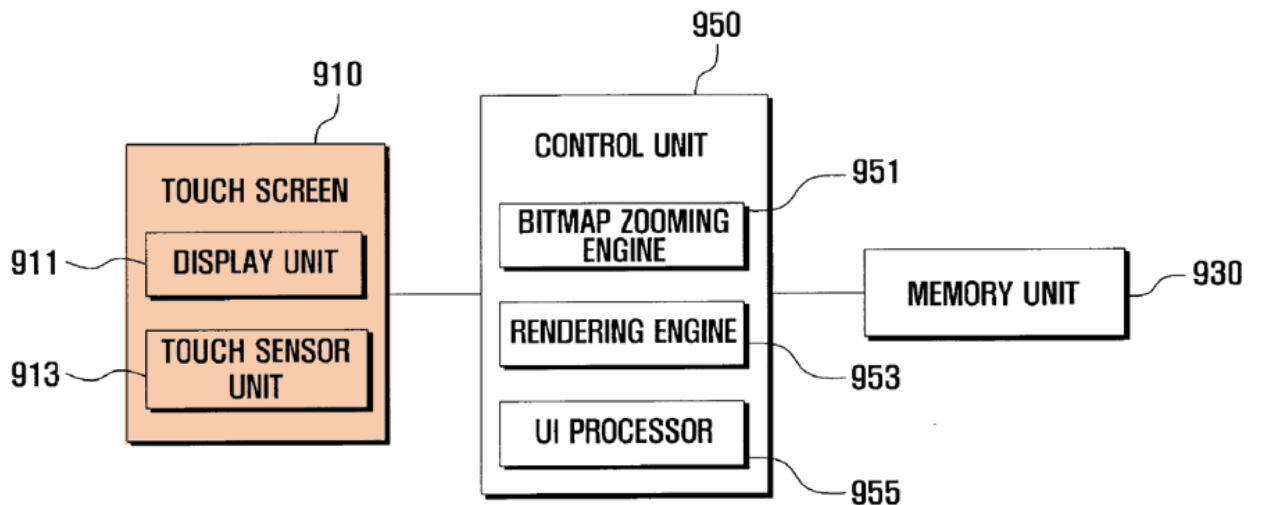
47. Kim teaches or renders [1.a] obvious. With reference to Figure 9, the Kim device includes “a memory unit 930.” APPLE-1004, [0122], [0127]-[0128]. The “memory unit 930 stores a plurality of applications required for controlling a general operation of the mobile device, and a plurality of applications required for controlling a zooming function based on a user gesture in the mobile device.” *Id.*, [0128]. The “applications are stored in an application storage region (not shown) in the memory unit 930.” *Id.* Kim further teaches that its methods are “realized in hardware or as software or computer code that can be stored in a recording medium ... so that the methods described herein can be executed by such software using a general purpose computer, or a special processor or in programmable or dedicated hardware.” *Id.*, [0144]. From this and other disclosures, a POSITA would have understood that the “applications,” “hardware,” “software,” and/or “computer code” of Kim include a series of instructions for performing the methods described in Kim, at least from Kim’s discussion of how the methods described in Kim are implemented. *Id.*



APPLE-1004, FIG. 9 (highlighted)

[1.b] a touch screen; and

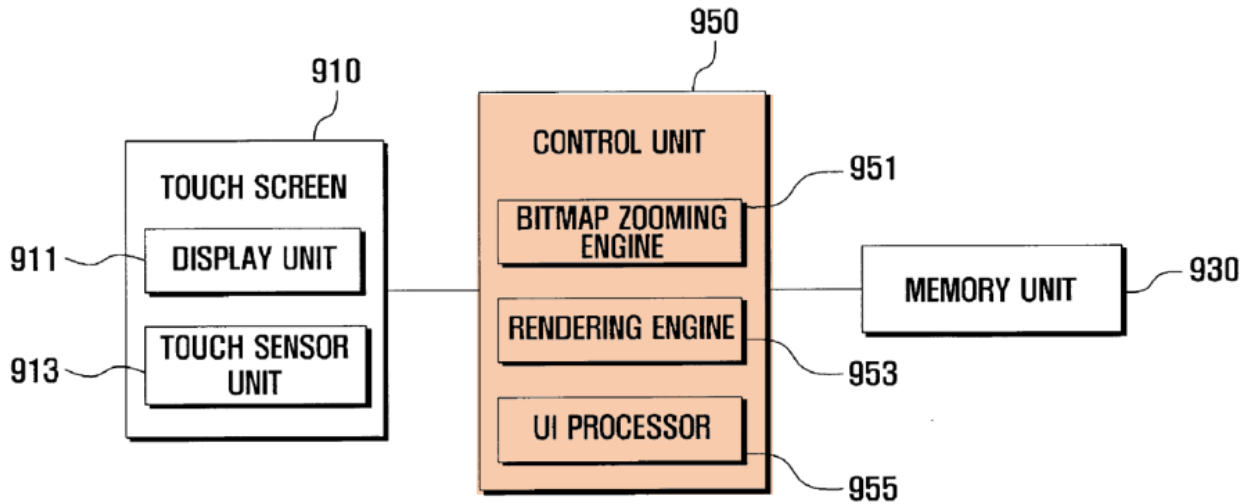
48. Kim teaches or renders [1.b] obvious. The Kim device includes “a touch sensitive surface such as a touch screen.” APPLE-1004, Abstract. With reference to Figure 9, Kim further shows in the inclusion of touch screen 910. *Id.*, [0122]-[0126], FIG. 9.



APPLE-1004, FIG. 9 (highlighted)

[1.c] 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 the instructions to:

49. Kim teaches or renders [1.c] obvious. With reference to Figure 9, the Kim device includes “a control unit 950,” which is shown to be in communication with the touch screen 910 and the memory unit 930. APPLE-1004, [0122], [0129]-[0135], FIG. 9. “The control unit 950 performs an overall control function associated with the mobile device and controls the flow of signals between components in the mobile device,” including “control[ling] final flows between the aforesaid components such as the touch screen 910 and the memory unit.” *Id.*, [0129]. Kim explains that the “control unit 950 controls particular operations in connection with the present invention.” *Id.*, [0135]. Thus, a POSITA would have understood that Kim’s control unit executes the instructions (e.g., computer code) stored in Kim’s memory to perform the methods described in Kim, at least based on Kim’s discussion of the functionality of Kim’s control unit 950. *Id.*, [0144] (“so that the methods described herein can be executed by such software using a general purpose computer”).

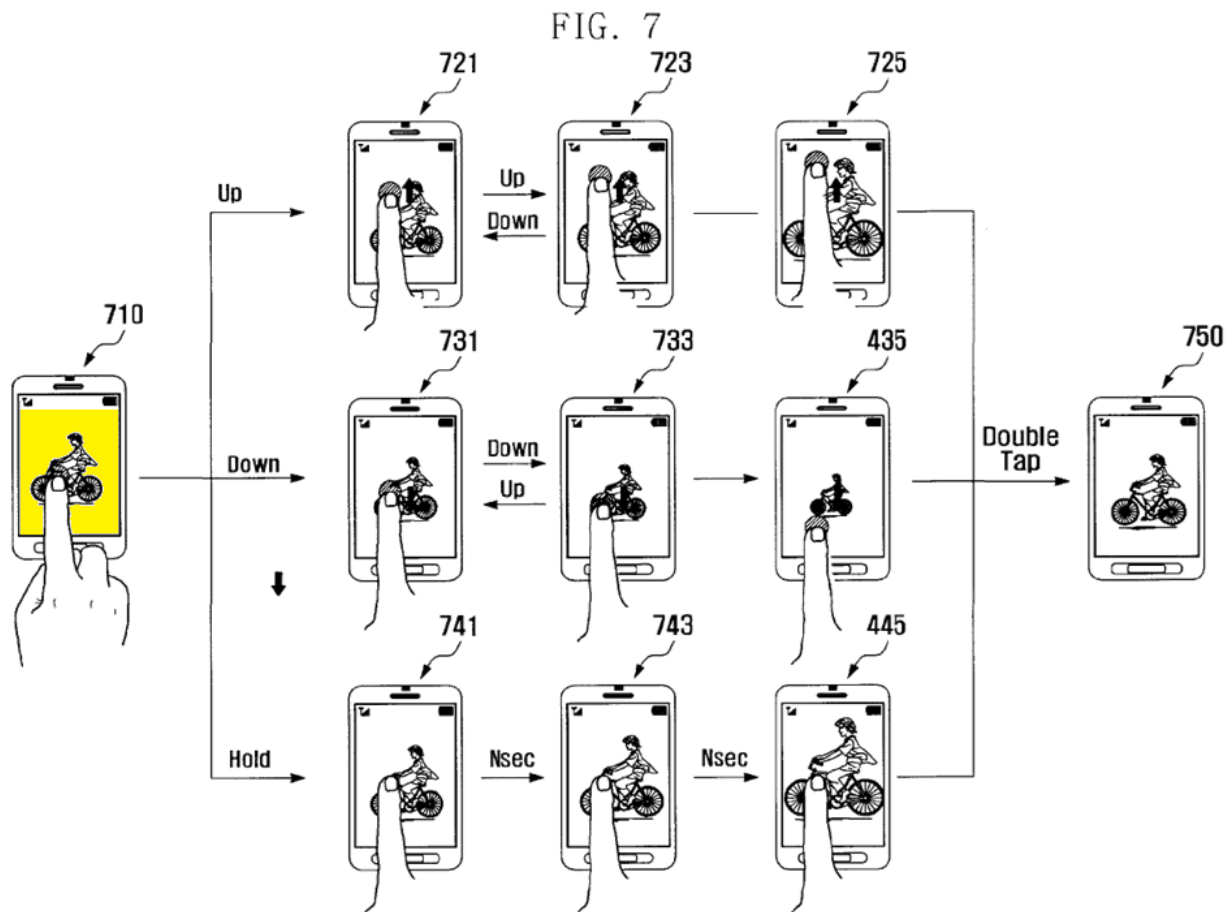


APPLE-1004, FIG. 9 (highlighted)

[1.d] display, via the touch screen, an object and at least a portion of an interface;

50. Kim teaches or renders [1.d] obvious. Kim describes several examples of the Kim device “display[ing] target data of specific content on the screen.” APPLE-1004, [0084]. For example, Kim describes “displaying some kind of display target data outputted while a specific application selected at a user’s request runs in the mobile device including any of a web page display mode based on a web browser function or environment, a photo data display mode based on a camera function or environment, [and] a text display mode based on a document viewer function or environment.” *Id.*, [0041], [0048]. In Figures 5-6, the Kim device displays “certain image data ... in an image viewer environment.” APPLE-1004, [0084], [0088], FIGS. 5-6; *see id.*, [0083], [0085]-[0087], [0089]-[0093]. Similarly, in Figure 7 (below), the Kim device “continuously offers a real-

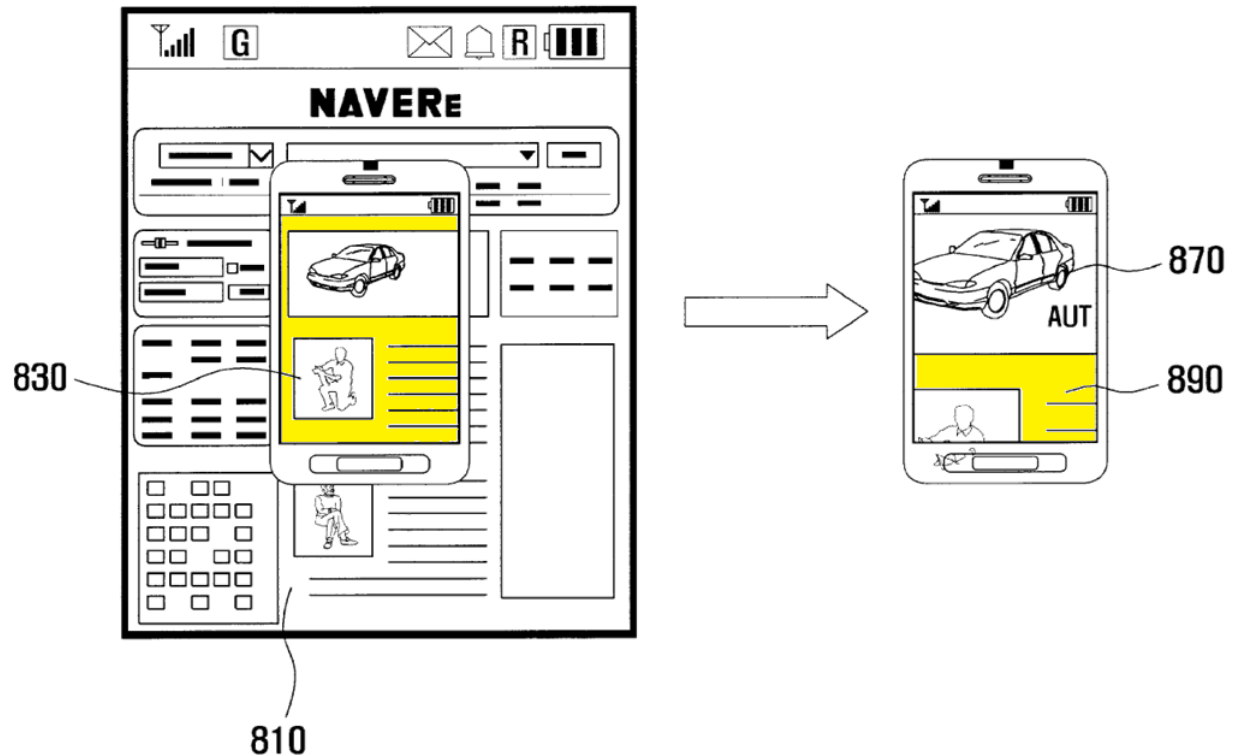
time, scaled-up bitmap image on the screen in response to the progress of the continuous move up event.” APPLE-1004, [0097], FIG. 7; *see id.*, [0094]-[0096], [0098]-[0107]. Figure 8 shows the Kim device displaying “target data” 870 in a bitmap image 810. APPLE-1004, [0109], FIG. 8; *see id.*, [0110]-[0114]. In each of these examples, the Kim device displays an object (e.g., person on bicycle in Figures 5-7 or car in Figure 8) surrounded by at least a portion of an interface (highlighted yellow in Figure 7).



APPLE-1004, FIG. 7 (highlighted)

51. In another example, Kim's Figure 8 (below) shows the display of an object (e.g., car, or text hyperlink) and a surrounding interface (highlighted yellow). APPLE-1004, [0109] ("target data"), FIG. 8.

FIG. 8



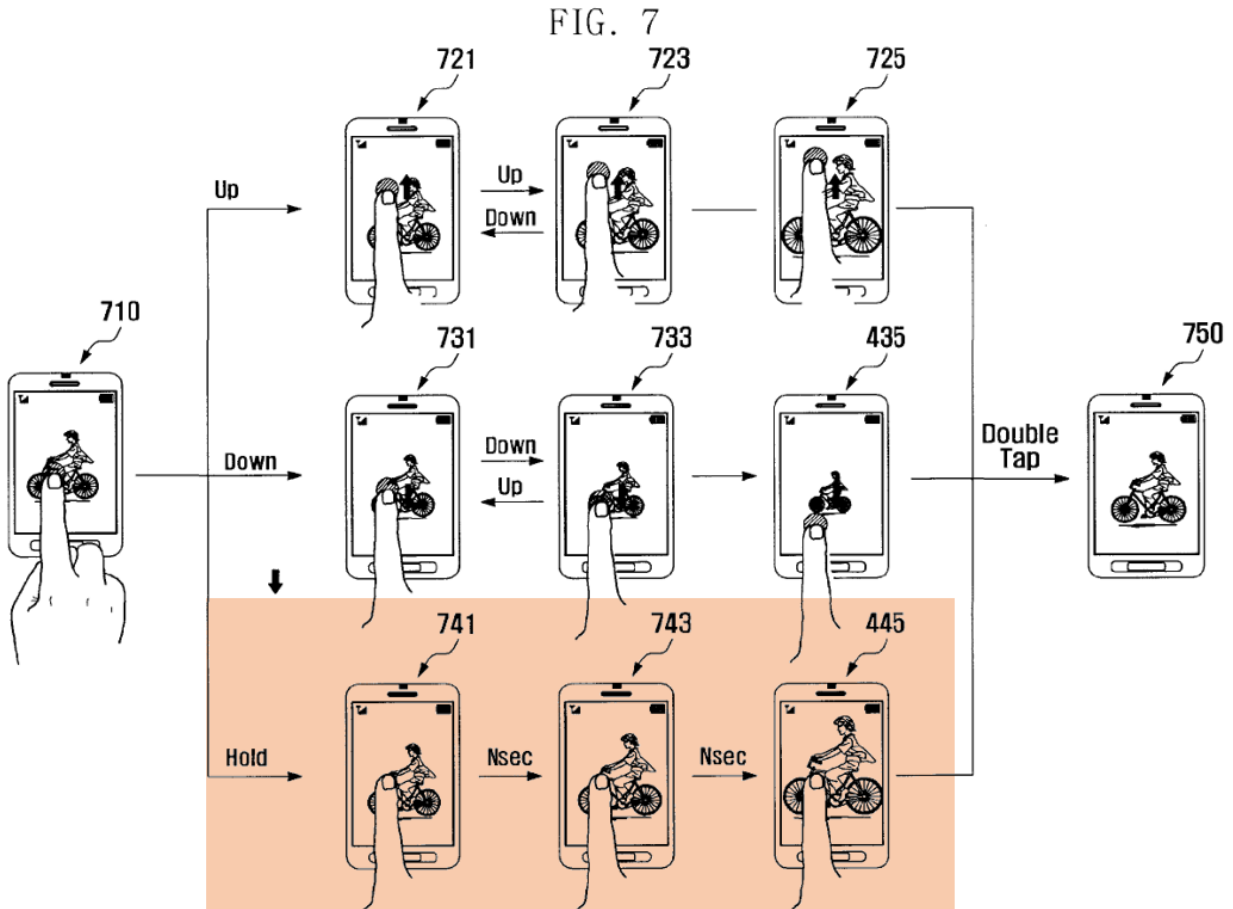
APPLE-1004, FIG. 8 (highlighted)

[1.e] detect a gesture on the touch screen on the object;

52. Kim teaches or renders [1.e] obvious. Kim describes various scenarios where a gesture is received on its touch screen. APPLE-1004, [0041]-[0045], [0048] ("detect the occurrence of an interrupt"), [0049] ("determine whether an interrupt is a long press input"), [0053] ("If detecting a user gesture, namely, if receiving an input event from a user through the touch screen...")

(“determines whether an input event is one of a move event (step 213), a hold event (step 231), and a double tap event (step 241)”), [0064], [0066], [0068], [0072], [0076], [0099], [0103]-[0104], [0106], [0130], FIGS. 7-8. In the highlighted example of Figure 7 (below), Kim describes “a continuous zoom-in action depending on a hold event.” APPLE-1004, [0104]. “As shown, a user can invoke the hold event at a selected point through a touch input such as tap and hold”—where the user taps and holds on the object (person on a bike), and in response to detecting this gesture, “the mobile device executes a continuous zoom-in function around the selected point.” *Id.* In addition to detecting gesture on the touch screen, Kim also teaches that the gesture is detected *on the object*. APPLE-1004, FIG. 7. As shown in Figure 7, the detected gesture is shown to be on the touch screen and *on the object* (the person on a bike). APPLE-1004, FIG. 7; *see also id.*, FIGS. 5-6. Indeed, a POSITA would have found it obvious that the gesture was detected on the object, as discussed above with respect to Kim.

§VII.A.1; APPLE-1004, [0104], FIGS. 5-7.



APPLE-1004, FIG. 7 (highlighted)

53. Kim provides various examples of gestures detected on the touch screen. In each of the examples discussed with respect to Kim's Figures 5-7, the Kim device detects a gesture on the touch screen on the object, at least as illustrated by the finger shown in each of those figures. APPLE-1004, FIGS. 5-7. With respect to Figure 7, for example, Kim states that "the mobile device can activate the zoom mode in response to a specific user gesture" by invoking a user input event at a selected point on display target data. APPLE-1004, [0095], [0085] ("a user may invoke a user input event for activating the zoom mode at a selected

point on display target data of content”) (“a user can invoke an input event at a desired position to be used as a central point of a zooming action”). Kim provides another example where a user gesture is received on an object (“a particular link assigned to certain text “AUTO”) at location (x1, y1) with respect to Figure 8.

APPLE-1004, [0108]-[0115], FIG. 8.

[1.f] when a magnitude of a duration of the gesture being detected on the touch screen on the object is less than a first magnitude threshold, perform an operation;

54. Kim teaches or renders [1.f] obvious. Kim describes “a double tap event for returning to an initial state with a basic display size after a zoom-in or zoom-out action.” APPLE-1004, [0031], [0064] (“if an input event is a double tap event as the result of determination in the step 241, the mobile device offers display target data with a predefined basic size”), [0099] (“in response to a double tap event, the mobile device displays target data with a given basic magnification value”), [0103], [0106]. In describing the touch as a tap, Kim discloses that the duration of each of the taps of the double tap gesture must be less than a threshold value associated with a longer-lasting (i.e., hold) event (the “first magnitude threshold”). In particular, from this disclosure and based on common knowledge of a tap, a POSITA would have understood that a double tap—a well known user input by the Critical Date—comprises two short duration taps in quick succession, which individually and collectively have a duration shorter than a threshold

duration associated with a more persistent gesture (e.g., a hold). APPLE-1005, [0057]-[0058], [0069]; APPLE-1009, 3 (“When I tell you to double-tap a specific location, you simply take your stylus and tap the item twice in a fairly short amount of time.”); APPLE-1012, [0072] (“A double tap action can be determined by the detection of a finger leaving the touch screen and immediately retouching the touch screen twice.”). The duration of each of the taps of the double tap gesture is “a duration of the gesture.” A POSITA would have understood that gesture recognition encompasses identification of multiple components of touch data. Newton teaches, for example, taking into account the duration of a gesture, a pattern of movement, and/or a distance by which fingers are separated. APPLE-1010, [0027], [0032]-[0035], [0054]. Marsden similarly teaches a process for “discern[ing] between the actions of tapping on the surface, resting on the surface, and pressing on the surface,” using “threshold for the touch signal” and “the rate of the rising edge of the touch signal to discern between a tap, a resting action, and a press.” APPLE-1011, [0065].

55. Consistent with this understanding, the specification of the ’114 patent states that “gesture input refers to the collection of inputs, flags, and signals by which a gesture is identified and parameterized,” that “[p]ossible gesture inputs may include, but are not limited to, touch event attributes (both initial and over time),” and that “[s]ome gestures may be able to be identified solely from initial

touch event attributes. Other gestures may only be identified after the gesture has been performed for some period of time.” APPLE-1001, 58:13-15, 58:19-20, 58:27-30. In Kim’s device, when a double tap event is generated, the “device displays target data with a given basic magnification value,” thus performing another operation. APPLE-1004, [0106].

[1.g] when the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than the first magnitude threshold, perform a scaling in connection with at least the object,

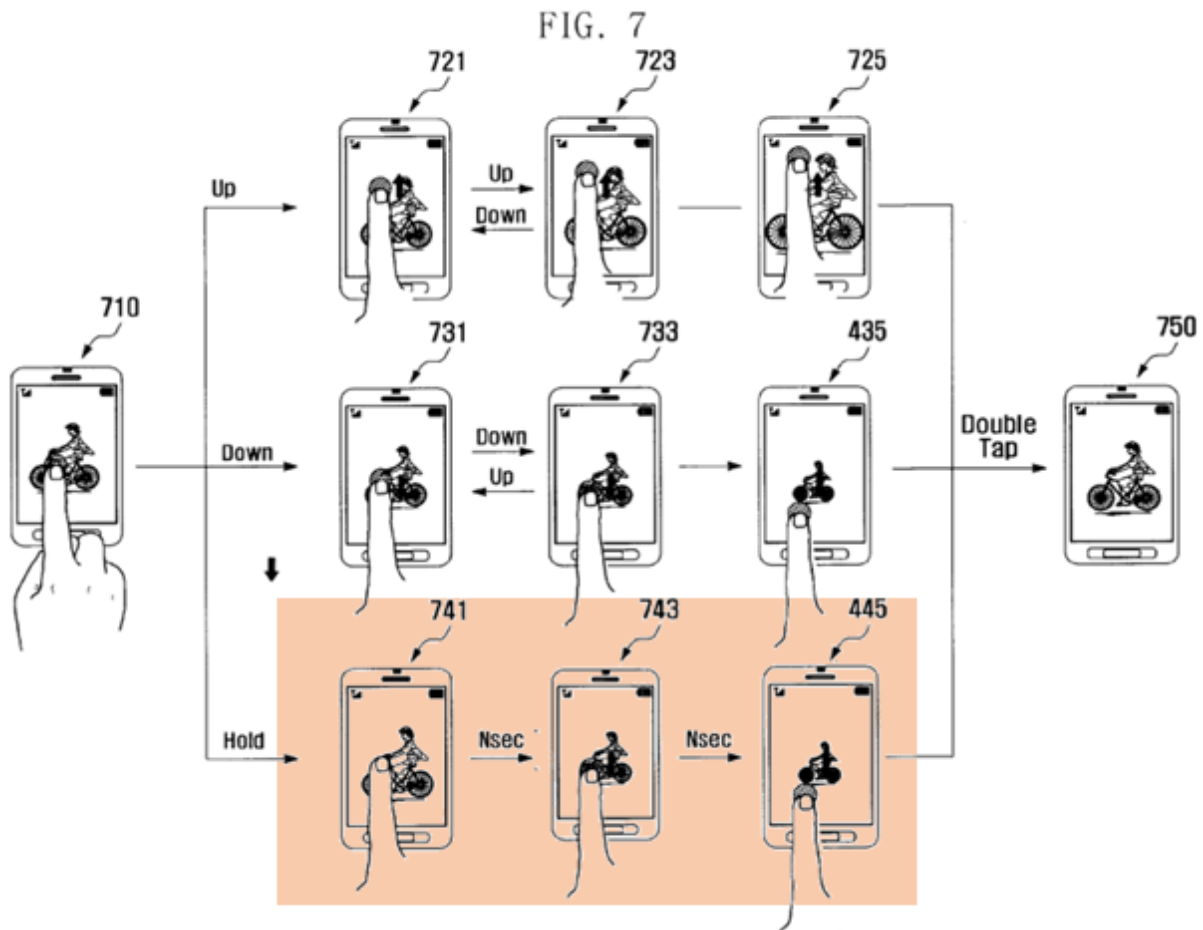
56. Kim teaches or renders [1.g] obvious. Kim explains that “reference numbers 741, 743 and 745 show a continuous zoom-in action depending on a hold event,” shown in the bottom path of Figure 7. APPLE-1004, [0104]. For such a hold event, a POSITA would have understood that the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than the first magnitude threshold (threshold value associated with a hold event), as was well known. APPLE-1013, [0050] (“A hold user interface event is generated by the user holding the finger on a touch screen or mouse pressed down for at least a predetermined time.”), [0052] (“When the first touch down event is received from the hardware, a timer may be started.”); APPLE-1014, [0049] (“if the user presses-and-holds without dragging for a short duration, such as one second”); APPLE-1015, 6:25-28 (“the user has selected a spot on the map and continued pressing for a long press period of time, such as by leaving their finger on the map for more

than several hundred milliseconds”), 8:66-9:4 (“At box 404, the process 400 determines whether sufficient time has expired for a long press period on the device. If such a time period has not yet expired, the process 400 determines whether the selected item exhibits long press functionality (box 405).”).

57. Kim explains that in “a zoom-in process, the mobile device continuously offers a real-time, scaled-up bitmap image on the screen in response to the duration of the hold event,” and when “the hold event is released from the screen, the mobile device displays an enlarge display target data around the selected point on the touch screen, depending on a zoom-in ratio of a bitmap image.” *Id.*, [0105]. Kim further explains that “the hold event is assigned to a continuous zoom-in function is an exemplary case only,” and “the hold event may be alternatively assigned to a continuous zoom-out function.” *Id.*, [0107].

58. Since “the hold event may be alternatively assigned to a continuous zoom-out function,” Kim teaches that bottom path of Figure 7 can be zoom-out instead of zoom-in, as illustrated in Figure 7 (modified), below (highlighted orange). APPLE-1004, [0107]. Figure 7 (modified) below shows a user gesture (hold event) detected on an object (person on bicycle) and the corresponding scaling operation performed on the object. APPLE-1004, FIG. 7.

59. Kim also refers to the request for the zoom mode being activated in response to a “long press input” and a “long press event.” APPLE-1004, [0031], [0034], [0042], [0049], [0085], [0139], Claim 2. A POSITA would have understood that Kim’s long press would be discriminated from a normal press based on the contact being maintained (without moving) for longer than a threshold duration. APPLE-1004, [0140]



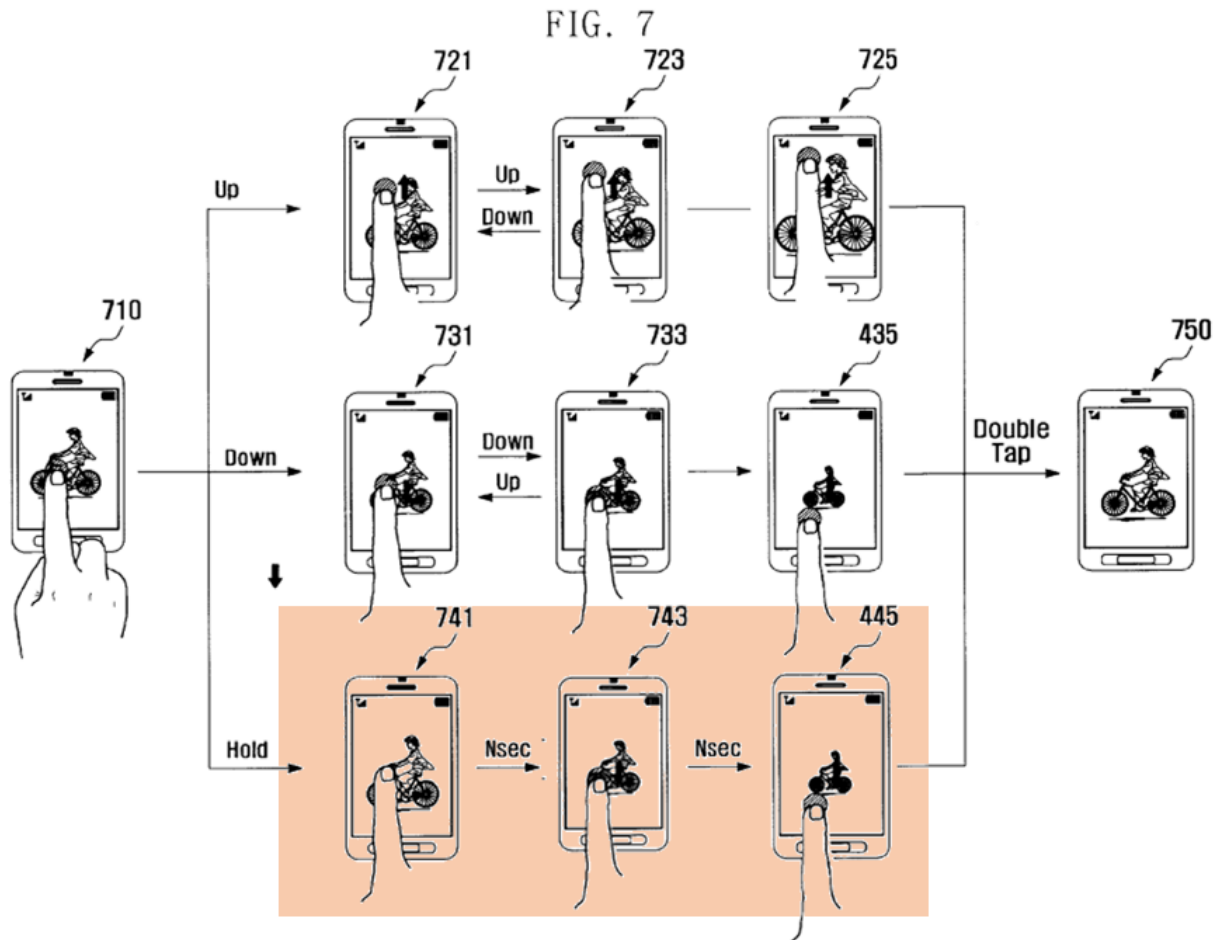
APPLE-1004, FIG. 7 (“hold” scenario modified)

[1.h] where the scaling includes a reduction in a size such that the size is inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the object; and³

60. Kim teaches or renders [1.h] obvious. A POSITA would have understood that a “continuous zoom-out function” on an object, such as the bike and bicyclist of Kim’s Figure 7, responsive to a persistent gesture (i.e., one with a constantly changing duration), is the equivalent of scale operation that includes a reduction in size of the object. Kim explains that, when the hold event is assigned to a zoom-in process, “the mobile device displays an enlarged display target data.” APPLE-1004, [0106]. “In such a zoom-in process, the mobile device continuously offers real-time, scaled-up bitmap image on the screen in response to the duration of the hold event.” *Id.*, [0105]. A POSITA would have understood that, when the hold event is assigned to a zoom-out process, the mobile device would instead display a scaled-down bitmap image on the screen in response to the duration of the hold event. APPLE-1004, [0031] (“a gesture refers to a touch-based user’s input motion which travels in a given pattern assigned to a zoom-in or zoom-out function in a zoom mode”), [0106] (“the hold event may be alternatively assigned

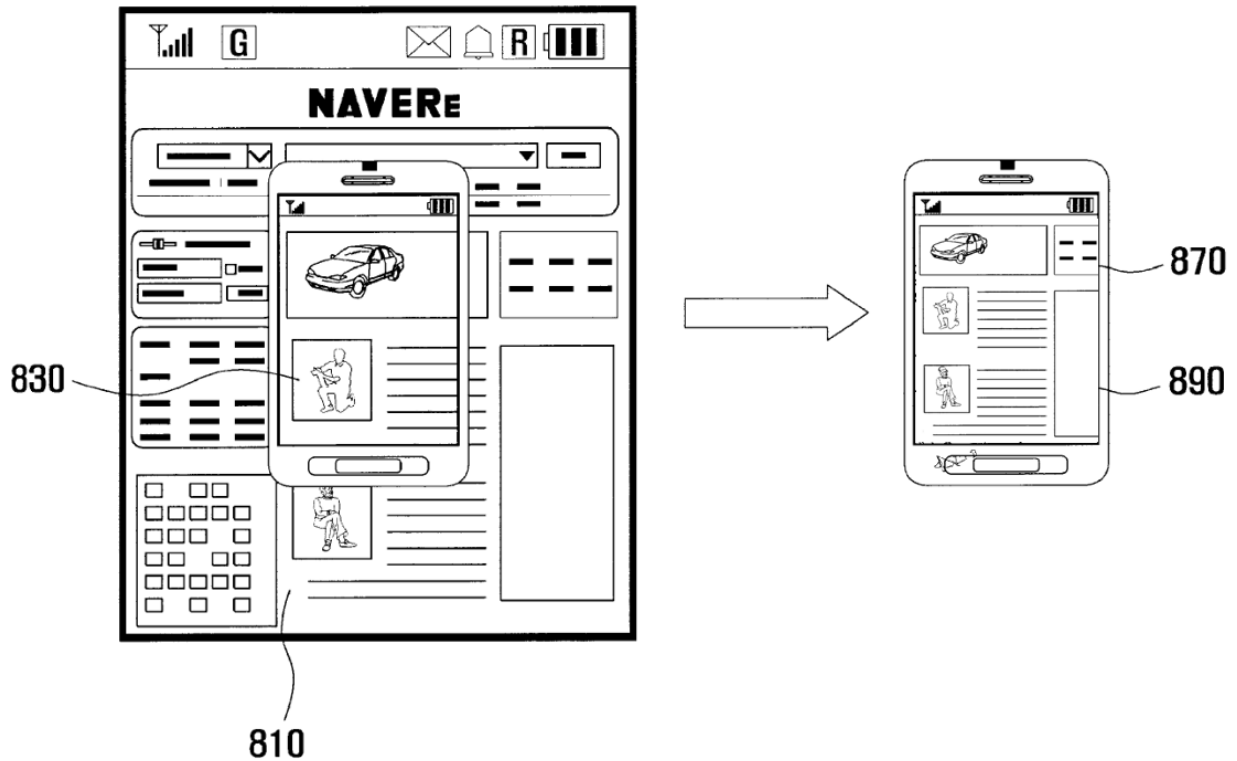
³ The discussion of [27.h] in §VII.A.3 is incorporated herein by reference to the extent “such that the size is inversely related to the magnitude of the duration...” is interpreted differently to require an inverse relationship between other values.

to a continuous zoom-out function”). Examples of a reduction in size shown below in modified Figures 7 and 8 of Kim below.



APPLE-1004, FIG. 7 (modified)

FIG. 8



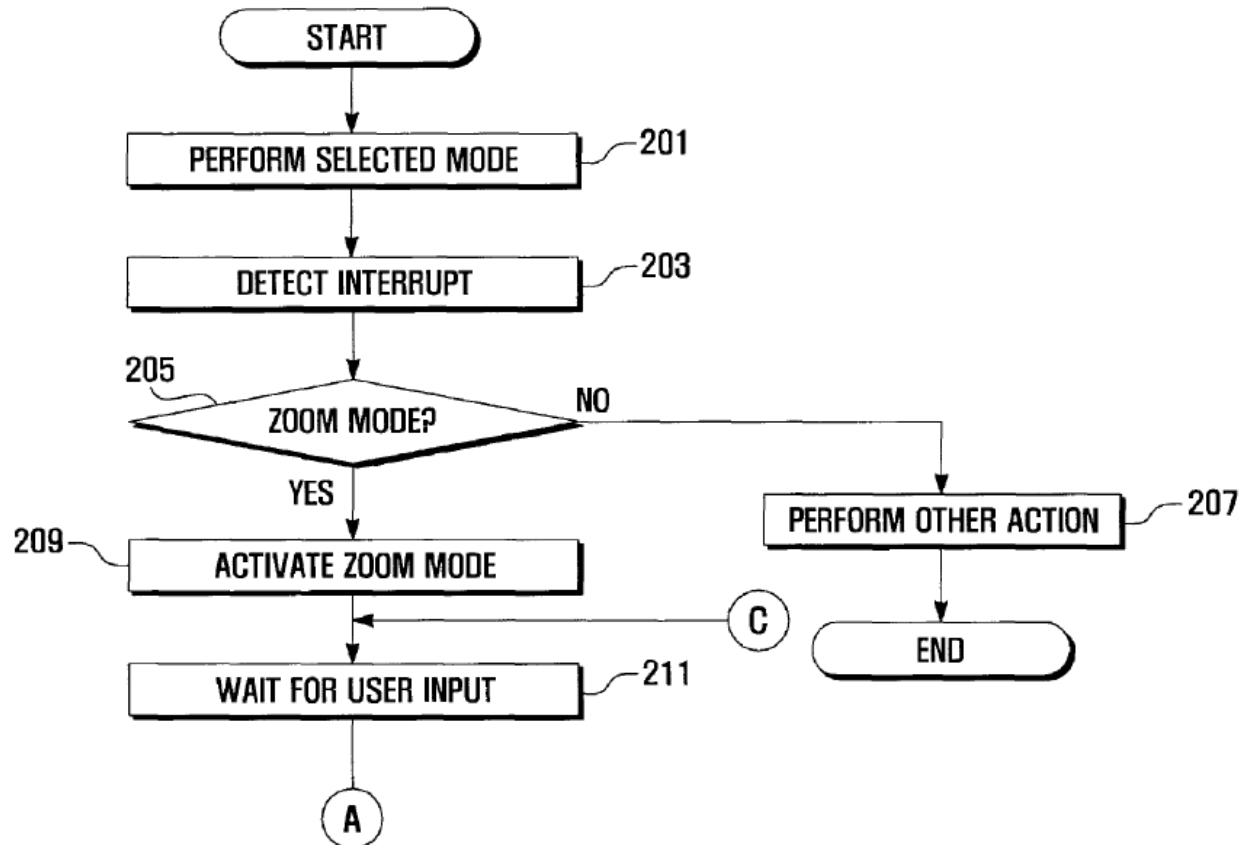
APPLE-1004, FIG. 8 (modified)

61. Additionally, Kim teaches that the zoom-out is “continuous”: in a zoom-out process, “the mobile device continuously offers a real-time, scaled-[down] bitmap image on the screen in response to the duration of the hold event.” APPLE-1004, [0105]. Thus, the longer the duration of the gesture (press and hold), the more the object is zoomed-out, or the more the size of the object is reduced. *Id.* In this way, the reduction in a size is inversely related to the magnitude of the duration of the gesture (the larger the duration of the gesture, the smaller the size). *Id.*

[1.i] when the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than a second magnitude threshold, perform another operation.

62. Kim teaches or renders [1.i] obvious. Kim describes “another operation” that is performed when the magnitude of the duration of the gesture is greater than a second magnitude threshold—specifically, a minimum threshold to be deemed an intentional touch.

63. In more detail, Kim explains that, “[i]f detecting a user gesture, namely, if receiving an input event from a user through the touch screen, the mobile device determines the type of an input event.” APPLE-1004, [0053]. Kim states that, “[i]f an interrupt is not a request for the activation of the zoom mode as the result of determination in step 205, the mobile device can perform some action in connection with an interrupt (step 207),” shown in Figure 2a below. APPLE-1004, [0050], FIG. 2a.



APPLE-1004, FIG. 2a

64. Kim additionally explains that if the mobile device determines that a detected user gesture does not correspond to any particular input event, “the mobile device disregards the received input event.” APPLE-1004, [0053]. Based on Kim’s disclosure of “disregarding” a received input, a POSITA would have understood that, for the detected interrupt (step 203) to be sufficient to cause the mobile device to “perform other action” (step 207), a magnitude greater than a minimum threshold value (e.g., the minimum threshold of an intentional user input) is relied upon, or rendered obvious, by Kim. APPLE-1004, FIG. 2a. Thus, when the magnitude of the gesture is greater than a second magnitude threshold,

the device “performs other action.” *Id.* For example, Kim discloses that, “if an interrupt is a request for a specific menu, the mobile device displays the requested menu.” APPLE-1004, [0050]. A POSITA would have understood that such an interrupt would be caused by a gesture that is longer in duration than the second threshold (as necessary to avoid the gesture from being disregarded), shorter than the duration necessary for a long press threshold, and a different gesture than a double tap gesture. APPLE-1016, [0033] (“cause the controller 130 to ignore or reject any touches that have a duration that is less than a predetermined minimum amount of time”); APPLE-1014, [0049] (“the graphic will appear if the user presses-and-holds without dragging for a short duration, such as one second”), [0115] (“if during a drag, the selection-point 31 becomes motionless for a long duration (5 seconds) within a function’s region, a pop-up menu of additional functions appears”).

3. Claim 27

[27.pre] An apparatus, comprising:

65. See discussion of [1.pre] (Ground 1A).

[27.a] at least one non-transitory memory storing instructions and a plurality of applications;

66. See discussion of [1.a] (Ground 1A).

[27.b] a touch screen; and

67. See discussion of [1.b] (Ground 1A).

[27.c] 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 the instructions to cause the apparatus to:

68. See discussion of [1.c] (Ground 1A).

[27.d] display an object on at least a portion of an interface;

69. See discussion of [1.d] (Ground 1A).

[27.e] detect a gesture via the touch screen on the object;

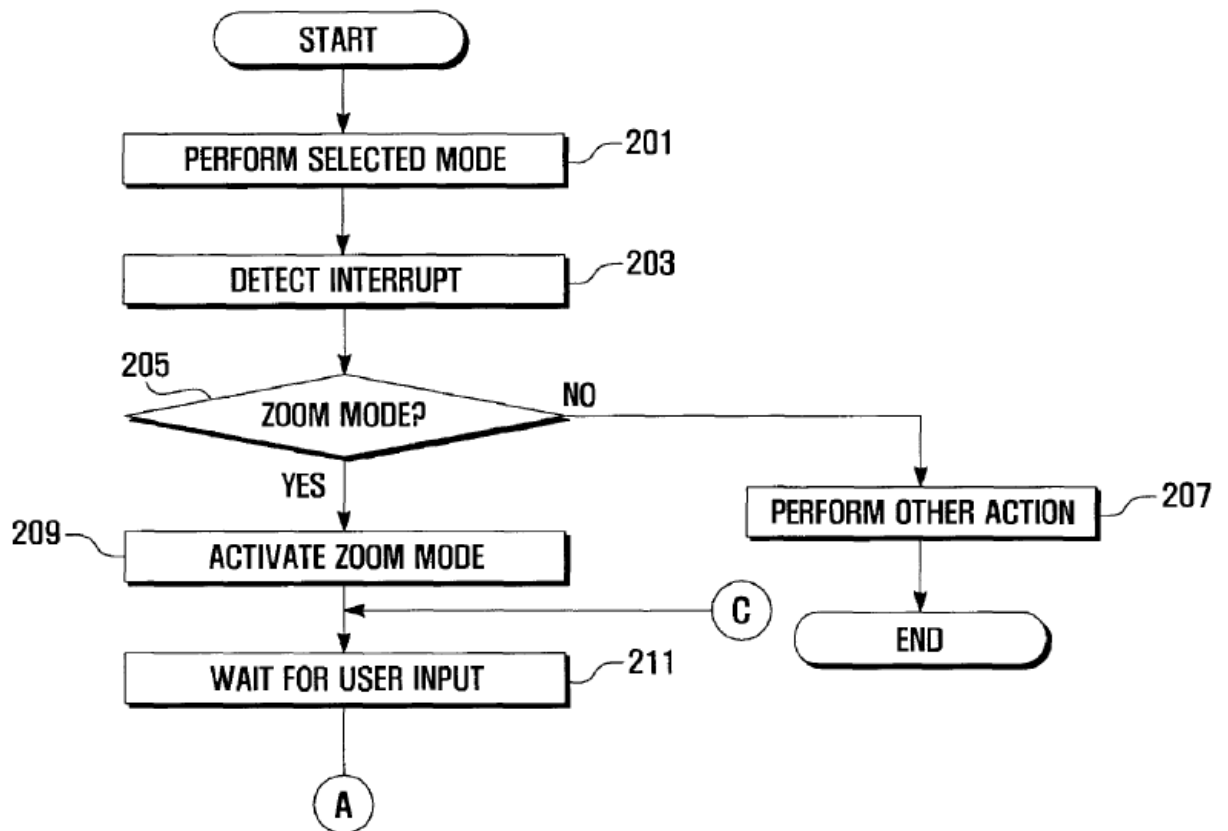
70. See discussion of [1.e] (Ground 1A).

[27.f] perform a first function in a first touch state;

71. Kim teaches or renders [27.f] obvious. Kim describes different modes, or touch states, with respect to its device (e.g., zoom mode, toggle mode) and corresponding functions performed in response to detected gestures in the respective mode. APPLE-1004, [0034]-[0035]. With reference to Figure 2A, Kim teaches receiving an interrupt (e.g., a gesture) at step 203 and then “determin[ing] whether [the] interrupt is a request for activating the zoom mode. APPLE-1004, [0048]-[0049], FIG. 2A. If not (e.g., if not in a zoom mode), “the mobile device can perform some action in connection with an interrupt (step 207).” *Id.*, APPLE-1004, [0050]. “For instance, if an interrupt is a request for a specific menu, the mobile device displays the requested menu. Also, if an interrupt is a selected input for a specific icon, the mobile device one of launches and enables one of an application and a function which is assigned to the selected icon.” *Id.* Additionally, “if the interrupt is a request for the toggle mode, the mobile device

performs appropriate actions in the toggle mode.” *Id.* In this way, Kim teaches performing a first function (“perform other action 207”) in a first touch state (i.e., when not in the “zoom mode” state). *Id.*

FIG. 2a



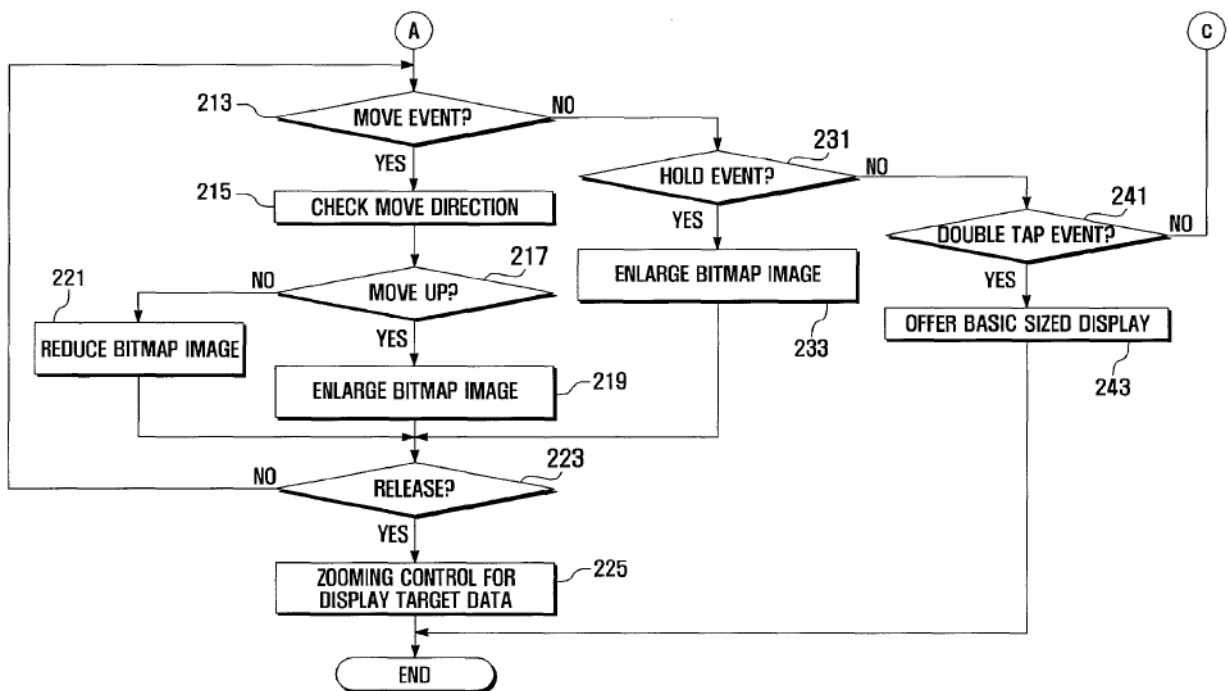
APPLE-1004, FIG. 2A

[27.g] perform a second function in a second touch state; and

72. Kim teaches or renders [27.g] obvious. In particular, Kim teaches performing different functions when in a different mode (e.g., zoom mode), or touch state. APPLE-1004, [0034]. Kim teaches that, “[i]f an interrupt is a request

for the activation of the zoom mode as a result of the determination in the step 205, the mobile device activates the zoom mode (step 209).” APPLE-1004, [0051]. In the zoom mode, the device can perform various actions. *Id.*, [0051]-[0064], FIG. 2B. For instance, with reference to Figure 2B, if the mobile device detects an “input event [that] is a double tap event,” “the mobile device offers display target data with a predefined basic size.” *Id.*, [0064], FIG. 2B. In this way, Kim teaches performing a second function (“offer basic sized display 243”) in a second touch state (i.e., when in the “zoom mode” state). *Id.*

FIG. 2b



APPLE-1004, FIG. 2B

[27.h] during detection of at least a portion of the gesture and based on a change in a duration of the gesture being detected via the touch screen on the object, perform a scale operation in connection with the at least portion of the interface, where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected via the touch screen on the object.

73. Kim teaches or renders [27.h] obvious. As discussed above for [1.g] and [1.h]⁴, in response to a press and hold gesture, the device performs a continuous zoom out (or, alternatively, a zoom in) function. APPLE-1004, [0104]-[0105], [0107], FIG. 7.

74. A POSITA would have understood that a “continuous zoom-out function” on an object, such as the bike and bicyclist of Kim’s Figure 7, is the equivalent of scale operation that includes a reduction in size of the object. Kim explains that, when the hold event is assigned to a zoom-in process, “the mobile device displays an enlarged display target data.” APPLE-1004, [0106]. “In such a zoom-in process, the mobile device continuously offers real-time, scaled-up bitmap image on the screen in response to the duration of the hold event.” *Id.*, [0105]. A POSITA would have understood that, when the hold event is assigned to a zoom-out process, the mobile device would instead display a scaled-down bitmap image on the screen in response to the duration of the hold event. APPLE-1004, [0031]

⁴ Where I reference discussions of other claim elements, I intend to incorporate by reference those earlier (or later) discussions into the particular section.

(“a gesture refers to a touch-based user’s input motion which travels in a given pattern assigned to a zoom-in or zoom-out function in a zoom mode”), [0106] (“the hold event may be alternatively assigned to a continuous zoom-out function”).

75. Additionally, Kim teaches that the zoom-out is “continuous”: in a zoom-out process, “the mobile device continuously offers a real-time, scaled-[down] bitmap image on the screen in response to the duration of the hold event.” APPLE-1004, [0105]. Thus, the longer the duration of the gesture (press and hold) (i.e., it has a continuous change in duration because the gesture persists), the greater the zoom-out, or the more the size is reduced. *Id.* Accordingly, to the extent this claim is interpreted to require a reduction in size, wherein the size is inversely related to the duration,⁵ Kim discloses the limitation.

76. In the alternative, if the phrase “where the scale operation includes a reduction in a size that is inversely related to the duration” is interpreted to mean that the *reduction* in size and the *duration* are inversely related (as opposed to the *size* and the *duration* being inversely related), then such an element would have at least been obvious. For example, Kim teaches that a double tap event (i.e., short duration) causes “enlarged display target data” to be “given basic magnification

⁵ This interpretation is consistent with element [1.h] (“a reduction in a size such that the size is inversely related to the magnitude of the duration”). §VII.A.2

value” (i.e., reduction in size), while a hold event (i.e., long duration) causes a zoom-out function (i.e., reduction in size). APPLE-1004, [0105]-[0107]. That is, in the case of a double tap, if the “enlarged display target data” was very large, the double tap would cause a large reduction in size, whereas, a hold that invokes a sustained and thus longer press would cause a zoom and correspondingly a decrease in the double-tap caused reduction in size. *Id.* In this way, the Kim scale operation responds to a change in a duration of the gesture (e.g., from a short duration to a long one) with a scale reduction that is inversely related to the length of touch duration. *Id.* Thus, the gesture with the longer duration of the gesture (long press) would cause a comparatively smaller reduction in size. *Id.* In this way, the *reduction* in a size is inversely related to the *duration* of the gesture (i.e., a gesture with a larger duration causes a smaller reduction in size than a gesture with a shorter duration).

77. In another alternative, if the phrase “where the scale operation includes a reduction in a size that is inversely related to the duration” is interpreted to mean that either the *reduction in size* or the *rate of reduction* and the *duration* are inversely related, then the element would still have at least been obvious. In particular, a POSITA would have understood that as an image is reduced in size (as zoomed out over time), fewer pixels are changed per step in the zoom-out animation (e.g., a step change from 100% to 99% zoom would change a larger

number of pixels than a step change from 2% to 1%). *Id.* In this way, the *reduction* in a size is inversely related to the *duration* of the gesture. *Id.*

78. Finally, while the example in Figure 7 shows an object (rider on a bicycle) with a surrounding interface that zooms in or zooms out in response to the press and hold, Figure 8 shows yet another example of the zoom-in or zoom-out effect. APPLE-1004, [0111]-[0115], FIG. 8. In particular, Figure 8 shows an example where, in response to a user gesture, a zooming action is performed on both an object (e.g., a car) and its surrounding interface (e.g., the remainder of the web page surrounding the object). *Id.* Thus, Kim's scale operation is performed in connection with the at least portion of the interface. *Id.*

4. Claim 27

79. This analysis of claim 27 presents "touch states" in a manner consistent with Smith's infringement contentions.

[27.pre] An apparatus, comprising:

80. *See* discussion of [1.pre] (Ground 1A).

[27.a] at least one non-transitory memory storing instructions and a plurality of applications;

81. *See* discussion of [1.a] (Ground 1A).

[27.b] a touch screen; and

82. *See* discussion of [1.b] (Ground 1A).

[27.c] 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 the instructions to cause the apparatus to:

83. See discussion of [1.c] (Ground 1A).

[27.d] display an object on at least a portion of an interface;

84. See discussion of [1.d] (Ground 1A).

[27.e] detect a gesture via the touch screen on the object;

85. See discussion of [1.e] (Ground 1A).

[27.f] perform a first function in a first touch state;

86. Kim teaches or renders [27.f] obvious. I have been informed that Patent Owner has alleged that turning on Bluetooth functionality in response to a tap input meets the requirements of [27.f], thereby equating a first touch state with a particular user input—that is, a tap.

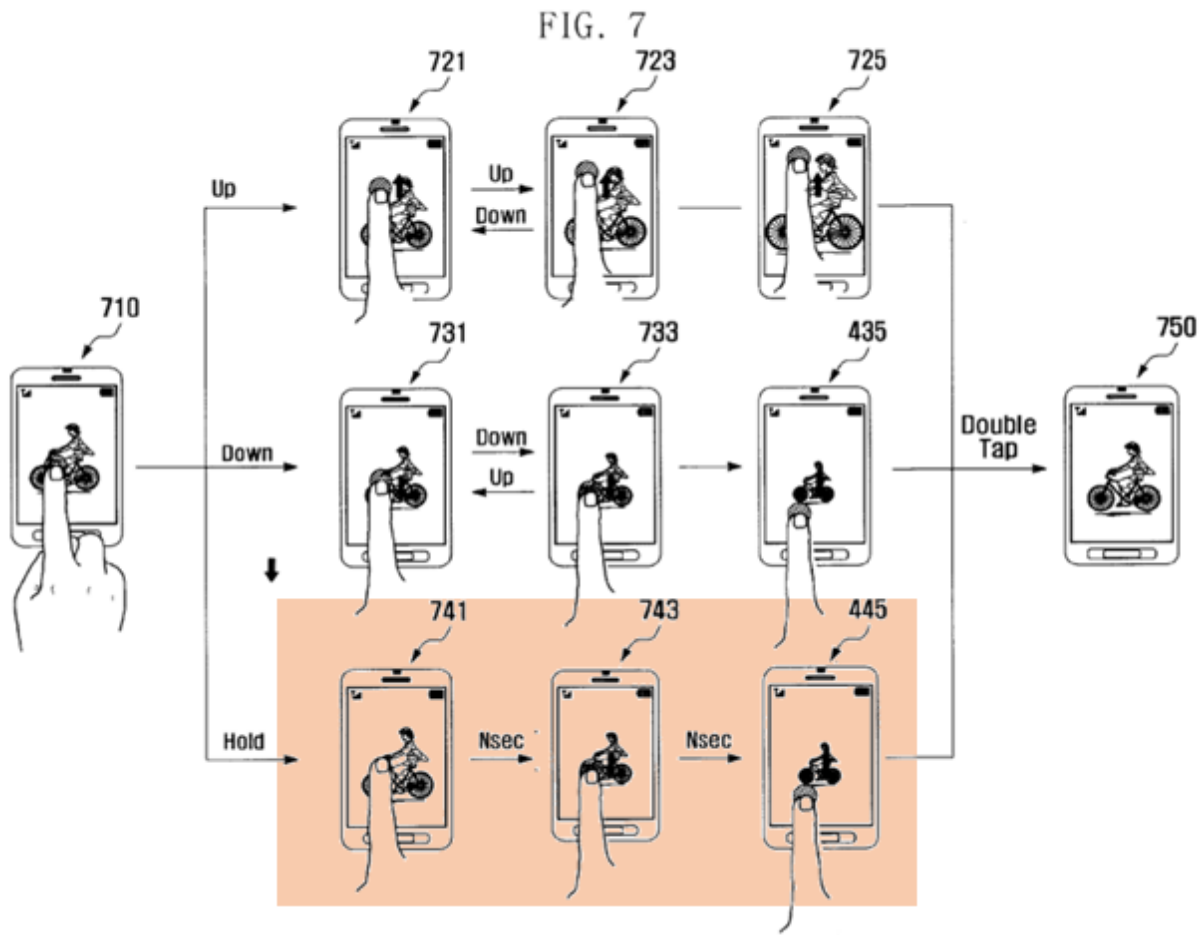
87. Kim provides a similar function performed in response to a user input. Specifically, “Reference numbers 721, 723 and 725 show a continuous zoom-in action depending on a move event,” shown in the top path of Figure 7. APPLE-1004, [0104]. Kim explains that “a user can continuously invoke a move up event at a selected point,” and “the mobile device executes a continuous zoom-in function around the selected point in response to the progress of a continuous move up event.” APPLE-1004, [0096], FIG. 7. In this way, Kim teaches performing a first function (continuous zoom-in function) in a first touch state (continuous move up event). *Id.*

[27.g] perform a second function in a second touch state; and

88. Kim teaches or renders [27.g] obvious. I have been informed that Patent Owner has alleged that causing a menu of icons to pop up in response to a long press input meets the requirements of [27.g], thereby equating a second touch state with a particular user input—that is, a long press.

89. Kim again teaches a similar function performed in response to a user input. Specifically, “reference numbers 741, 743 and 745 [445] show a continuous zoom-in action depending on a hold event,” shown in the bottom path of Figure 7 of Kim. APPLE-1004, [0104]. Kim explains that in “a zoom-in process, the mobile device continuously offers a real-time, scaled-up bitmap image on the screen in response to the duration of the hold event.” *Id.*, [0105]. Kim further explains that “the hold event is assigned to a continuous zoom-in function is an exemplary case only,” and “the hold event may be alternatively assigned to a continuous zoom-out function.” *Id.*, [0107].

90. Since “the hold event may be alternatively assigned to a continuous zoom-out function,” Kim teaches that bottom path of Figure 7 can be zoom-out instead of zoom-in, as illustrated in Figure 7 (modified), below (highlighted orange). APPLE-1004, [0107]. In this way, Kim teaches performing a second function (zooming out, or, alternatively, zooming in) in a second touch state (press and hold). *Id.*



APPLE-1004, FIG. 7 (“hold” scenario modified)

[27.h] during detection of at least a portion of the gesture and based on a change in a duration of the gesture being detected via the touch screen on the object, perform a scale operation in connection with the at least portion of the interface, where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected via the touch screen on the object.

91. Kim teaches or renders [27.h] obvious. As discussed above for [27.g], in response to a press and hold gesture, the device performs a continuous zoom out (or, alternatively, a zoom in) function. APPLE-1004, [0104]-[0105], [0107], FIG. 7.

92. A POSITA would have understood that a “continuous zoom-out function” on an object, such as the bike and bicyclist of Kim’s Figure 7, is the equivalent of scale operation that includes a reduction in size of the object. Kim explains that, when the hold event is assigned to a zoom-in process, “the mobile device displays an enlarged display target data.” APPLE-1004, [0106]. “In such a zoom-in process, the mobile device continuously offers real-time, scaled-up bitmap image on the screen in response to the duration of the hold event.” *Id.*, [0105]. A POSITA would have understood that, when the hold event is assigned to a zoom-out process, the mobile device would instead display a scaled-down bitmap image on the screen in response to the duration of the hold event. APPLE-1004, [0031] (“a gesture refers to a touch-based user’s input motion which travels in a given pattern assigned to a zoom-in or zoom-out function in a zoom mode”), [0106] (“the hold event may be alternatively assigned to a continuous zoom-out function”).

93. Additionally, Kim teaches that the zoom-out is “continuous”: in a zoom-out process, “the mobile device continuously offers a real-time, scaled-[down] bitmap image on the screen in response to the duration of the hold event.” APPLE-1004, [0105]. Thus, the longer the duration of the gesture (press and hold), the greater the zoom-out, or the more the size is reduced. *Id.* Therefore, to the extent this claim is interpreted to require a reduction in size, wherein the size is inversely related to the duration, Kim discloses the limitation. *Id.*

94. In the alternative, if the phrase “where the scale operation includes a reduction in a size that is inversely related to the duration” is interpreted to mean that the *reduction* and the *duration* are inversely related (as opposed to the *size* and the *duration* being inversely related), then such an element would have at least been obvious. For example, Kim teaches that a double tap event (i.e., short duration) causes “enlarged display target data” to be “given basic magnification value” (i.e., reduction in size), while a hold event (i.e., long duration) causes a zoom-out function (i.e., reduction in size). APPLE-1004, [0105]-[0107]. That is, in the case of a double tap, if the “enlarged display target data” was very large, the double tap would cause a large reduction in size, where, a long press that was relatively short would cause a smaller reduction in size. *Id.* Thus, the scale operation is responsive to a change in a duration of the gesture (e.g., from a short duration to a long one). *Id.* Thus, the gesture with the longer duration of the gesture (long press), would cause a comparatively smaller reduction in size. *Id.* In this way, the *reduction* in a size is inversely related to the *duration* of the gesture (i.e., a gesture with a larger duration causes a smaller reduction in size than a gesture with a shorter duration). *Id.*

95. In another alternative, if the phrase “where the scale operation includes a reduction in a size that is inversely related to the duration” is interpreted to mean that either the *reduction in size* or the *rate of reduction* and the *duration*

are inversely related, then such an element would have at least been obvious. In particular, a POSITA would have understood that as an image is reduced in size (as zoomed out over time), fewer pixels are changed per step in the zoom-out animation (e.g., a step change from 100% to 99% zoom would change a larger number of pixels than a step change from 2% to 1%). *Id.* In this way, the *reduction* in a size is inversely related to the *duration* of the gesture. *Id.*

96. Finally, while the example in Figure 7 shows an object (rider on a bicycle) with a surrounding interface that zooms in or zooms out in response to the press and hold, Figure 8 shows yet another example of the zoom-in or zoom-out effect. APPLE-1004, [0111]-[0115], FIG. 8. In particular, Figure 8 shows an example where, in response to a user gesture, both an object (e.g., a car) and its surrounding interface (e.g., the remainder of the web page surrounding the object), a zooming action is performed on both the object and its surrounding interface. *Id.* Thus, Kim's scale operation is performed in connection with the at least portion of the interface. *Id.*

5. Claim 32

[32.pre] An apparatus, comprising:

97. See discussion of [27.pre] (Ground 1A).

[32.a] at least one non-transitory memory storing instructions and a plurality of applications;

98. See discussion of [27.a] (Ground 1A).

[32.b] a touch screen; and

99. See discussion of [27.b] (Ground 1A).

[32.c] 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 the instructions to:

100. See discussion of [27.c] (Ground 1A).

[32.d] display, via the touch screen, a first object and at least one other object;

101. Kim teaches or renders [32.d] obvious. As discussed above with respect to [1.d], Kim displays a first object. APPLE-1004, FIG. 7. To the extent the device shown in Kim's Figure 7 is interpreted as a single object, a POSITA would have found the display of at least one other object obvious and understood Figure 7 to simply be an example. Indeed, Figure 8 of Kim, for example, shows more than one object (e.g., a car, text or hyperlink such as hyperlink assigned to text "AUTO", a person). APPLE-1004, FIG. 8. Thus, a POSITA would have found the display of any number of objects at least to be obvious.

[32.e] detect at least one component of a gesture on the touch screen on the first object;

102. Kim teaches or renders [32.e] obvious. See discussion of [1.e] (Ground 1A). A POSITA would have understood or found obvious that the gesture detected on Kim's device's touch screen includes a plurality of time-related components, including, for example, how long the gesture has been maintained at any point in time during or at cessation of the gesture. A POSITA would have

readily understood the implication of these time-related components in Kim to reveal or render obvious various types of gestures based on the length of the gesture. See APPLE-1004, [0034] (“*long press*”), [0061] (“duration of the hold event”), [0105] (“*duration* of the hold event”), [0139] (“*long press*”).

103. Additionally, a POSITA would have understood from Kim’s discussion of interacting with content on the touch screen that the gesture can be received (and detected) and any number of objects displayed on the screen. For instance, with respect to Figure 8, Kim describes, among the displayed objects, a hyperlink with the text “AUTO” (“first object”),⁶ and a POSITA would have found it obvious for a user to input a gesture on the hyperlink (“first object”), while not inputting the gesture on other objects (e.g., car, person). APPLE-1004, [0113]-[0114], FIG. 8.

[32.f] when a first component of the gesture is detected where a magnitude of a duration of the gesture being detected on the touch screen on the first

⁶ Note that while Figure 8 does not show the hyperlink “AUTO” text as visible in the viewport 830, Kim explains, “Assume that the viewport 830 contains a particular link assigned to certain text ‘AUTO’ with coordinate (x1, y1). Furthermore, assume that web display data is enlarged several times as indicated by a reference number 870.” APPLE-1004, [0114].

object is less than a first magnitude threshold, perform a function upon a cessation of the gesture being detected;

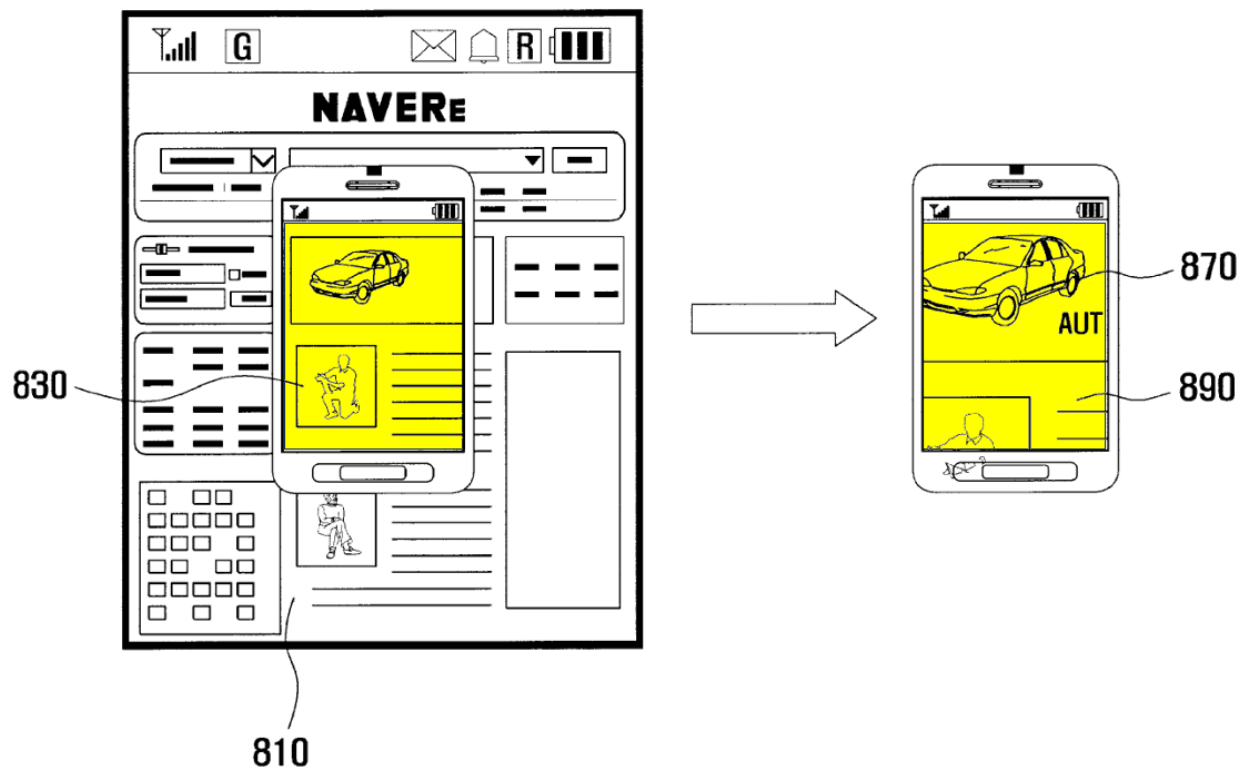
104. Kim teaches or renders [32.f] obvious for at least the same reasons as discussed for [1.f]. In particular, when a first component of the gesture (time-related component when a double tap event is generated) is detected where the duration detected on the touch screen is less than a first magnitude (e.g., threshold associated with a hold event), Kim's device performs a function (displays target data with a given basic magnification value). *See* discussion of [1.f]. Moreover, a POSITA would have understood that, where the function occurs after timeout, shortly after cessation of the gesture (when the user releases his finger), the function is performed upon cessation of the gesture being detected. APPLE-1004, [0103] ("Thereafter, a user can invoke a double tap event on the screen. Then, *in response to a double tap event*, the mobile device displays target data with a given basic magnification value...").

[32.g] when a second component of the gesture is detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is greater than the first magnitude threshold, perform a scale operation for the at least one other object,

105. Kim teaches or renders [32.g] obvious for at least the same reasons as discussed for [1.g]. In particular, when a second component of the gesture (time-related component when a hold event is detected) is detected where the duration detected on the touch screen is greater than the first magnitude (threshold

associated with a hold event), Kim's device performs a function (a zoom or scale operation). *See* discussion of [1.g] (Ground 1A). Moreover, a POSITA would have understood that the scale operation is performed on content displayed on the screen, including both the first object (hyperlink) and the at least one other object (e.g., car, person). Figure 8 of Kim (below) shows an example showing how all of the displayed content (highlighted) is scaled in response to the gesture. APPLE-1004, FIG. 8. That is, multiple objects (text, car, person) are simultaneously scaled. *Id.*

FIG. 8



APPLE-1004, FIG. 8 (highlighted)

[32.h] where the scale operation includes a reduction in a size so as to be inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the first object; and

106. See discussion of [1.h].

[32.i] when a third component of the gesture is detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is greater than a second magnitude threshold, perform another function.

107. Kim teaches or renders [32.i] obvious for at least the same reasons as discussed for [1.i]. In particular, when a third component of the gesture (e.g., time-related component when the interrupt is detected) is detected where the duration detected on the touch screen is greater than a second magnitude (the minimum threshold of an intentional user input), Kim's device performs another function ("perform other action"). APPLE-1004, FIG. 2a; see discussion of [1.i].

B. Ground 1B: Claims 34, 59, and 64 are obvious over Kim in view of Lang

1. Lang

108. Lang is directed to a "user interface for small computing devices." APPLE-1006, Abstract. It explains that the "small computing devices include personal digital assistants (PDAs), smart watches, mobile telephones, and the like." *Id.*, [0015]. "An exemplary smart watch 102 embodying the present invention" includes input elements, such as "touch sensitive elements." *Id.*, [0016]-[0017]. The smart watch 102 also includes a "display screen 118" that "may be touch-

sensitive, thereby providing activation signals to the smart watch 102 when the display screen 118 is contacted by the user.” *Id.*, [0020].

2. The Kim-Lang Combination

109. Kim describes a mobile device with “a touch sensitive surface such as a touch screen” that can “include, but is not limited to, a device selected from the group comprising a portable multimedia player (PMP), a digital broadcasting player, a personal digital assistant (PDA), a game console, a smart phone, a music player, a car navigation system, *and any other kinds of portable and handheld devices.*” APPLE-1004, [0118]. To further understand what other types of devices Kim’s mobile device could be implemented as, a POSITA would have naturally looked at other similar discussions of portable devices.

110. For example, Lang describes a “user interface for small computing devices,” explaining that “[s]uch small computing devices include personal digital assistants (PDAs), *smart watches*, mobile telephones, and the like.” APPLE-1006, Abstract, [0015] (emphasis added). Like Kim, Lang describes a smart watch having “touch sensitive elements,” such as a “display screen 118” that is “touch-sensitive, thereby providing activation signals to the smart watch 102 when the display screen 118 is contacted by the user.” APPLE-1006, [0017], [0020]. Thus, a POSITA would have understood that “any other kinds of portable and handheld devices” include smart watches, as Lang teaches. APPLE-1006, [0015].

111. Given that Lang’s smart watch includes a touch-sensitive display screen, a POSITA would have understood that Kim’s device could be implemented as such a smart watch, e.g., allowing a user to invoke zooming functions, discussed above in §VII.A. In addition, given that Kim contemplates implementing its device in “any other kinds of portable and handheld devices” a POSITA would have reasonably expected success in implementing Kim’s device as a smart watch. Combining Kim with Lang (the Kim-Lang Device, or “KLD”) would have been obvious at least because such a combination would have merely involved (1) combining prior art elements according to known methods to yield predictable results and (2) the use of known technique to improve similar devices (methods, or products) in the same way.

3. Claim 34

[34.pre] A method, comprising:

112. To the extent the preamble is limiting, the KLD renders [34.pre] obvious. In particular, as discussed with respect to claim 1, the Kim device (apparatus) performs a various functions. §VII.A. For at least the same reasons, the KLD performs a method, as discussed in more detail with respect to [34.e]-[34.j]. See discussion of [1.pre] (Ground 1A).

[34.a] at an apparatus including a watch with

113. The KLD renders [34.a] obvious. Kim explains that its mobile device can be “any other kinds of portable and handheld devices.” APPLE-1004, [0118].

As discussed above in §VII.B.2, a POSITA would have found it obvious to implement the KLD as a smart watch, as taught by Lang.

[34.b] at least one non-transitory memory storing a plurality of applications,

114. See discussion of [1.a] (Ground 1A).

[34.c] a touch screen, and

115. See discussion of [1.b] (Ground 1A).

[34.d] one or more processors in communication with the at least one non-transitory memory and the touch screen:

116. See discussion of [1.c] (Ground 1A).

[34.e] displaying, via the touch screen, an object and at least a portion of an interface;

117. See discussion of [1.d] (Ground 1A).

[34.f] detecting a gesture on the touch screen on the object;

118. See discussion of [1.e] (Ground 1A).

[34.g] based on a magnitude of a duration of the gesture detected on the touch screen on the object being detected to be less than a first magnitude threshold, performing an operation;

119. See discussion of [1.f] (Ground 1A).

[34.h] based on the magnitude of the duration of the gesture detected on the touch screen on the object being detected to be greater than the first magnitude threshold, performing a scaling in connection with at least the object,

120. See discussion of [1.g] (Ground 1A).

[34.i] where the scaling includes a reduction in a size such that the size is inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the object; and

121. See discussion of [1.h] (Ground 1A).

[34.j] based on the magnitude of the duration of the gesture detected on the touch screen on the object being detected to be greater than a second magnitude threshold, performing another operation.

122. See discussion of [1.i] (Ground 1A).

4. Claim 59

[59.pre] A method, comprising:

123. See discussion of [34.pre] (Ground 1B).

[59.a] at an apparatus including a watch with

124. See discussion of [34.a] (Ground 1B).

[59.b] at least one non-transitory memory storing a plurality of applications,

125. See discussion of [34.b] (Ground 1B).

[59.c] touch screen, and

126. See discussion of [34.c] (Ground 1B).

[59.d] one or more processors in communication with the at least one non-transitory memory and the touch screen:

127. See discussion of [34.d] (Ground 1B).

[59.e] displaying an object on at least a portion of an interface;

128. See discussion of [27.d] (Ground 1A).

[59.f] detecting a gesture via the touch screen on the object;

129. See discussion of [27.e] (Ground 1A).

[59.g] performing a first function in a first touch state;

130. See discussion of [27.f] (Ground 1A).

[59.h] performing a second function in a second touch state; and

131. See discussion of [27.g] (Ground 1A).

[59.i] based on a detection of at least a portion of the gesture and based on a change in a duration of the gesture being detected via the touch screen on the object, performing a scale operation in connection with the at least portion of the interface, where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected via the touch screen on the object.

132. See discussion of [27.h] (Ground 1A).

5. Claim 64

[64.pre] A method, comprising:

See discussion of [34.pre] (Ground 1B).

[64.a] at an apparatus including a watch having

See discussion of [34.a] (Ground 1B).

[64.b] at least one non-transitory memory storing a plurality of applications,

See discussion of [34.b] (Ground 1B).

[64.c] a touch screen, and

See discussion of [34.c] (Ground 1B).

[64.d] one or more processors in communication with the at least one non-transitory memory and the touch screen:

See discussion of [34.d] (Ground 1B).

[64.e] displaying, via the touch screen, a first object and at least one other object;

See discussion of [32.d] (Ground 1A).

[64.f] detecting at least one component of a gesture on the touch screen on the first object;

See discussion of [32.e] (Ground 1A).

[64.g] in response to a first component of the gesture being detected where a magnitude of a duration of the gesture being detected on the touch screen on the first object is detected to be less than a first magnitude threshold, performing a function upon a cessation of the gesture being detected;

See discussion of [32.f] (Ground 1A).

[64.h] in response to a second component of the gesture being detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is detected to be greater than the first magnitude threshold, performing a scale operation for the at least one other object,

See discussion of [32.g] (Ground 1A).

[64.i] where the scale operation includes a reduction in a size so as to be inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the first object; and

See discussion of [32.h] (Ground 1A).

[64.j] in response to a third component of the gesture being detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is detected to be greater than a second magnitude threshold, performing another function.

See discussion of [32.i] (Ground 1A).

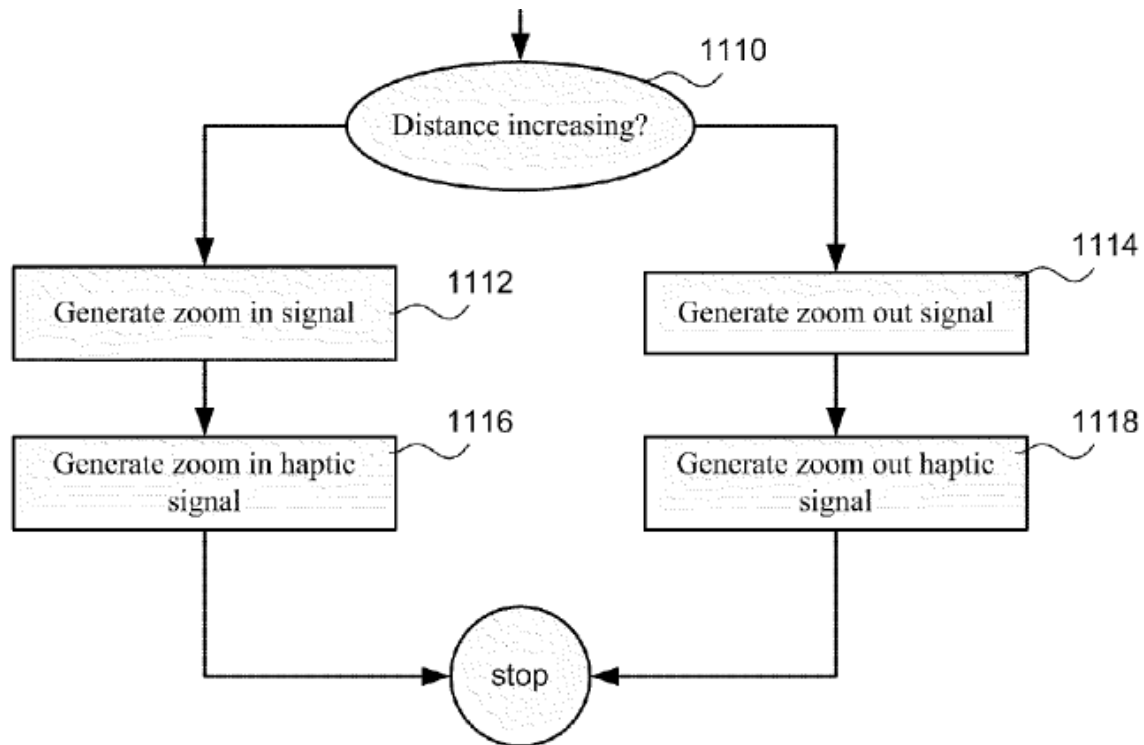
C. Ground 1C: Claim 30 is obvious over Kim in view of Burrough

1. Burrough

133. Burrough describes “[m]ethods and systems for processing touch inputs.” APPLE-1007, Abstract. Burrough explains that “research has shown that providing [a] multi-touch surface with the ability to provide physical (haptic) feedback makes the multi-touch experience even more efficient and realistic to the user.” *Id.*, [0008]. Burrough describes portable electronic device 100 (e.g., “small form factor electronic devices”) that includes a “haptic actuator 136” that “creates haptic response” (“vibro-tactile response”). *Id.*, [0037], [0047], FIGS. 1A-1E. The haptic actuator can be a “piezoelectric actuator assembly,” for example, that “generates force F directly proportional to voltage V applied to the haptic actuator.” *Id.*, [0055]-[0056]. The actuator applies “a physical force to the touch sensitive surface that results in a vibro-tactile response ... felt by the user.” *Id.*, [0013], [0033].

134. With reference to Figure 11 (below), Burrough describes “a zoom gesture method” that is “performed on multi-touch multi-haptic surface.” APPLE-1007, [0079], FIG. 11. In particular, Burrough explains that, in response to zoom gesture, “the haptic devices nearest the touch point are set to active mode in order to provide a vibro-tactile response ... to each of the fingers during the gesture.” *Id.* In some scenarios, different haptic signals are provided for a zoom-out signal versus a zoom-in signal. *Id.*, [0080]; *see id.*, [0081]. As shown below, following

any zoom signal (1112 or 1114), a haptic signal is generated (1116, 1118). *Id.*, FIG. 11.



APPLE-1007, FIG. 11 (excerpted)

2. The Kim-Burrough Combination

135. Kim describes a mobile device that “has a touch sensitive surface such as a touch screen” that provides “a touch-based zooming control method.”

APPLE-1004, Abstract. Like many devices with touch sensitive surfaces, however, Kim does not whether tactile feedback is provided to a user, and instead leaves this detail to POSITAs well aware of haptic feedback by the Critical Date.

For instance, Lacroix explains:

Electronic device manufacturers strive to produce a rich interface for users. Conventional devices use visual and auditory cues to provide feedback to a user. In some interface devices, kinesthetic feedback (such as active and resistive force feedback) and/or tactile feedback (such as vibration, texture, and heat) is also provided to the user, more generally known collectively as “haptic feedback.” Haptic feedback can provide cues that enhance and simplify the user interface. Specifically, vibration effects, or vibrotactile haptic effects, may be useful in providing cues to users of electronic devices to alert the user to specific events, or provide realistic feedback to create greater sensory immersion within a simulated or virtual environment.

APPLE-1008, [0003]. The lack of a physical keyboard or tactile means for providing input was known to create challenges for users, especially where a user is visually distracted or visually impaired, lighting is poor, and screen sizes are small. APPLE-1017, [0003], [0004] (“In some situations, it is desirable for the user to be able to locate keys without looking at the touch screen. For example, when driving, a user may prefer to locate and select a menu item or locate and dial numbers on a software keypad using only tactile feedback in order to maintain visual contact with the road. As another example, a vision-impaired user may be unable to operate a touch screen having key locations without tactile feedback.”); APPLE-1018, [0003] (“Portable electronic devices have shrunk while providing ever more functionality. Because the devices are smaller, users can more easily carry them, and use them in a variety of situations where larger, less portable electronic devices could be more difficult to use. An effect of the decrease in

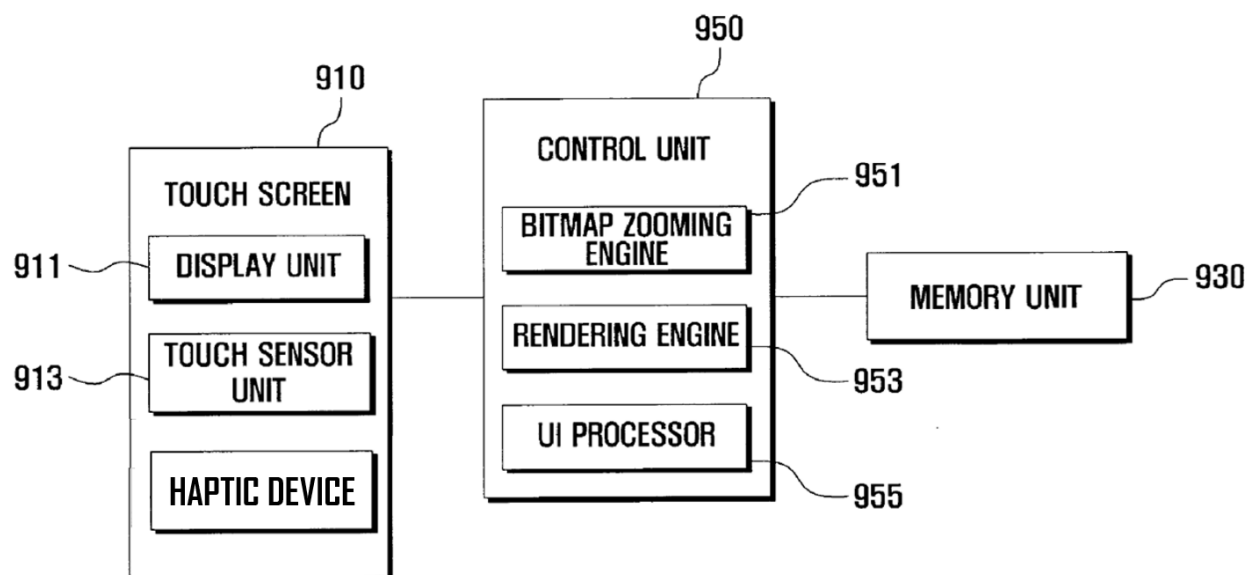
size of electronic devices has been a decrease in the size of input mechanisms for providing instructions to the electronic device, and a decrease in the size of displays or screens used to navigate the electronic devices and to access the electronic device functions.”), [0004] (“For some users, including for example users with poor vision, or users in dark environments (e.g., at night with little or no electronic device backlight), the reduction in size of the input mechanism and display inconveniences the users and may prevent the users from properly using the electronic device, thus leading to frustration.”). As such, a POSITA would have been motivated to look to references, such as Burrough, which teaches the use of haptic feedback in conjunction with touch screens. APPLE-1007, Abstract.

136. Indeed, Burrough recognizes that “physical sensation can not be provided by a conventional multi-touch system thereby forcing the user to visually locate the home key thereby making keyboard use less efficient and fatiguing.” APPLE-1007, [0008]. “[R]esearch has shown that providing the multi-touch surface with the ability to provide physical (haptic) feedback makes the multi-touch experience even more efficient and realistic to the user.” *Id.* Thus, Burrough teaches a portable device that provides haptic feedback in response to user input, including zooming operations. *Id.*, [0079]-[0081], FIG. 11. The haptic feedback is provided by way of a haptic device (e.g., piezoelectric actuator

assembly) in the portable device that provides vibro-tactile response to the user.

Id., [0013], [0033], [0037], [0047], [0055]-[0056], FIGS. 1A-1E

137. A POSITA would therefore have been motivated to include a haptic device and use of haptic feedback, as Burrough teaches and suggests, in the Kim device to provide haptic feedback to improve the user experience with the touch screen. In particular, a POSITA would have been motivated to provide haptic feedback to a user invoking a zoom operation, as Burrough teaches, for example, to provide an indication as to whether a zoom-in or a zoom-out gesture has been invoked. APPLE-1007, [0080]. As shown in Figure 1B of Burrough, the haptic actuator 136 is in communication with a micro-controller 132 and processor 106 of the portable device. APPLE-1007, [0046] (“Although micro-controller 132 is shown as a component separate from processor 106, it is contemplated that functions carried out by micro-controller 132 can in fact be performed by processor 106.”), [0047] (“cause microcontroller 132 to active at least haptic actuator 136”), FIG. 1B. In the Kim-Burrough Device (“Kim-Burrough-Device”), the haptic device would be in communication with the processor (“control unit 950”) of Kim, such that the control unit 950 could control haptic device and its generating of haptic responses on the touch screen, as illustrated below in modified Figure 9 from Kim.



APPLE-1004, FIG. 9 (modified)

138. A POSITA would have reasonably expected success in implementing Kim's device with a haptic device since Burrough teaches the use of including haptic devices in a portable devices, such as Kim's device. Combining Kim with Burrough would have been obvious at least because such a combination would have merely involved (1) combining prior art elements according to known methods to yield predictable results and (2) the use of known technique to improve similar devices (methods, or products) in the same way.

3. Claim 30

[30.pre] An apparatus, comprising:

139. See discussion of [1.pre] (Ground 1A).

[30.a] at least one non-transitory memory storing instructions and a plurality of applications;

140. See discussion of [1.a] (Ground 1A).

[30.b] a touch screen;

141. See discussion of [1.b] (Ground 1A).

[30.c] a vibrator; and

142. The Kim-Burrough-Device renders [1.c] obvious. As discussed above in §VII.C.2, the Kim-Burrough-Device further includes a haptic device (e.g., piezoelectric actuator assembly), which provides a “vibro-tactile response,” and thus is a vibrator. APPLE-1007, [0013], [0033], [0037], [0047], [0055]-[0056], FIGS. 1A-1E; §VII.C.2.

[30.d] one or more processors in communication with the at least one non-transitory memory, the touch screen, and the vibrator, wherein the one or more processors execute the instructions to cause the apparatus to:

143. The Kim-Burrough-Device renders [30.d] obvious with respect to the one or more processors in communication with memory and the touch screen for at least the same reasons as discussed for [1.c]. As discussed above in §VII.C.2, the Kim-Burrough-Device further includes a vibrator in communication with the control unit of Kim. §VII.C.2.

144. As discussed above in §VII.C.2, the Kim-Burrough-Device further includes a vibrator in communication with the one or more processors. §VII.C.2.

[30.e] display a first object and at least one other object;

145. See discussion of [32.d] (Ground 1A).

[30.f] detect a gesture on the touch screen on the first object;

146. See discussion of [1.e] (Ground 1A).

[30.g] during detection of at least a portion of the gesture before a completion thereof is detected, perform a scale operation, based on a change in a duration of the gesture being detected on the touch screen on the first object, on at least the at least one other object,

147. The Kim-Burrough-Device renders [30.g] obvious for at least the same reasons as discussed for [32.g]-[32.h]. As discussed above with respect to [32.g]-[32.h], the Kim-Burrough-Device performs a scale operation on a displayed object other than the first object subject to the gesture based on a change in a duration of a detected gesture on the first object (e.g., the gesture continuing and thus increasing in duration). As discussed with respect to [30.e], a POSITA would have found the display of more than one object to at least be obvious.

Additionally, a POSITA would have found it obvious that the scale operation would be performed on the first object and the at least one other object, as clearly shown in Figure 8 of Kim. APPLE-1004, FIG. 8.

148. Moreover, Kim teaches that the scaling operation (e.g., continuous zoom action) occurs during the tap and hold event: “the mobile device executes a continuous zoom-in [or zoom-out] function around the selected point at given periodic intervals while the hold event continues.” APPLE-1004, [0104], [0107].

Thus, a POSITA would have understood that the scale operation is performed during detection of at least a portion of the gesture before a completion thereof is detected.

[30.h] where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected on the touch screen on the first object; and

149. See discussion of [32.h] (Ground 1A).

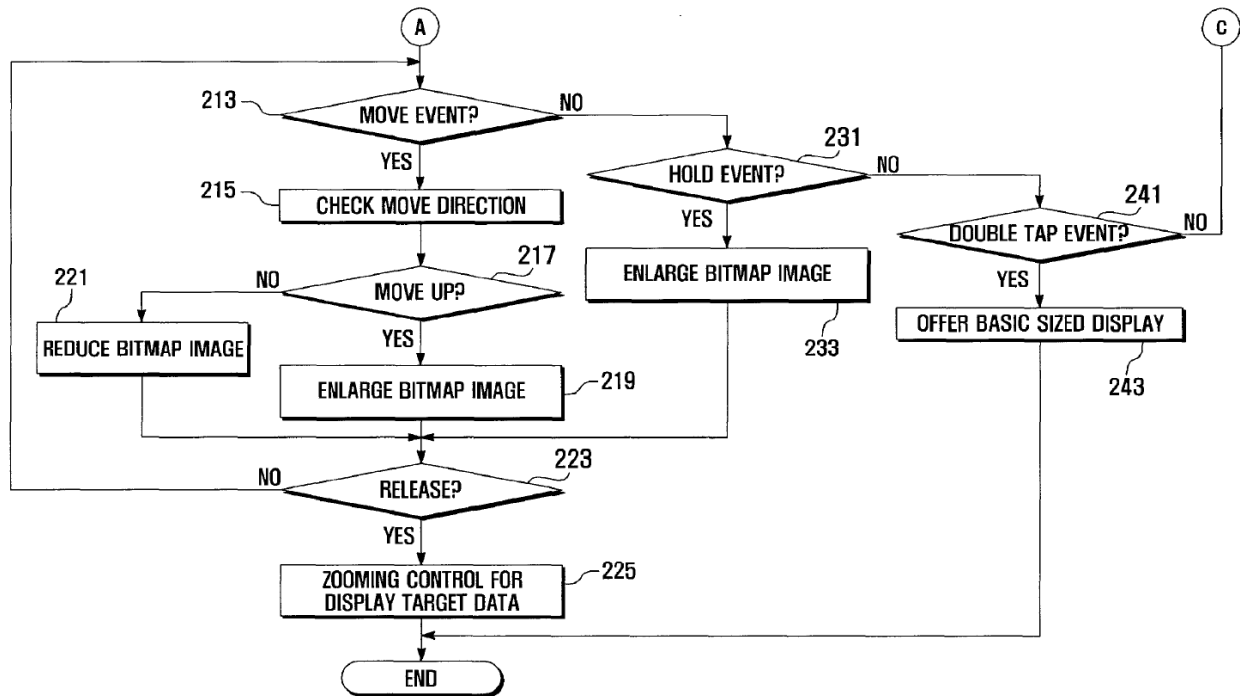
[30.i] after at least a portion of the scale operation, vibrate the apparatus and perform an action.

150. The Kim-Burrough-Device renders [30.i] obvious. Kim explains that after a hold event that causes a zoom-in (or zoom-out) process, “[t]hereafter, a user can invoke a double tap event on the screen,” causing the device to display “target data with a given basic magnification value.” APPLE-1004, [0106], FIG. 7; *see id.*, FIG. 2b. Specifically, the flowchart in Kim's FIG 2b (below) shows zoom enlargement at 233 in response to a Hold event at 231. APPLE-1004, FIG. 2b. After the zoom enlargement at 233, the flowchart discloses that a “RELEASE?” decision is made at 223, and if the answer is “NO”, the chart loops back to decision 213 that leads to “DOUBLE TAP EVENT” decision at 241. *Id.*

Therefore, Kim discloses detecting the “DOUBLE TAP EVENT” and performing an action (“OFFER BASIC SIZED DISPLAY” 243) after the scaling operation at 233. Thus, after at least a portion of the scale operation, the device performs an

action (displays data with a given basic magnification). *Id.*, [0106], FIG. 2b, FIG.

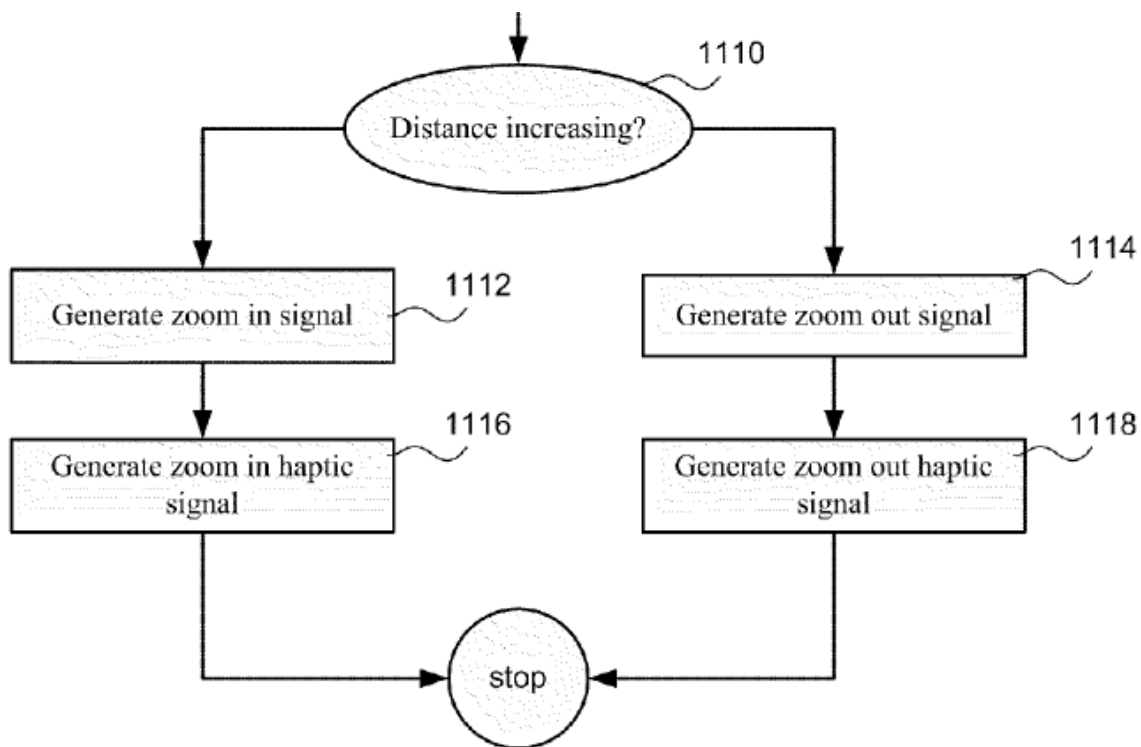
7.



APPLE-1004, FIG. 2b

151. As discussed above in §VII.C.2, the Kim-Burrough-Device includes a haptic device (e.g., piezoelectric actuator assembly) that provides a “vibro-tactile response.” APPLE-1007, [0013], [0033], [0037], [0047], [0055]-[0056], FIGS. 1A-1E, FIG. 11; §VII.C.2. As shown in Figure 11 (excerpted below), Burrough teaches “generate a zoom out haptic signal” or “generate a zoom in haptic signal” and thus inspiring haptic vibration after steps 1112/1114, which result in a zoom in or zoom out operation, thus rendering obvious haptic vibration after zooming or scaling. APPLE-1004, [0080], FIG. 11. That is, a POSITA would have

understood from Burrough that a haptic signal (1116/1118 of Figure 11) follows any zooming signal in the Kim-Burrough-Device, and therefore, the zooming action that occurs to “offer basic sized display” (243 in FIG. 2b of Kim) in the Kim-Burrough-Device would be accompanied by a haptic output, as Burrough teaches.



APPLE-1007, FIG. 11 (excerpted)

D. Ground 1D: Claim 62 is obvious over Kim in view of Lang and Burrough

1. The Kim-Lang-Burrough Combination

152. The Kim-Lang Device (KLD) is discussed above in §VII.B.2. For at least the same reasons discussed in §VII.C.2, it would have further been obvious to

a POSITA to modify the KLD to include a haptic device and use of haptic feedback, as taught and suggested by Burrough.

2. Claim 62

[62.pre] A method, comprising:

153. See discussion of [34.pre] (Ground 1B).

[62.a] at an apparatus including a watch

154. See discussion of [34.a] (Ground 1B).

[62.b] having at least one non-transitory memory storing a plurality of applications,

155. See discussion of [34.b] (Ground 1B).

[62.c] a touch screen,

156. See discussion of [34.c] (Ground 1B).

[62.d] a vibrator, and

157. See discussion of [30.c] (Ground 1C).

[62.e] one or more processors in communication with the at least one non-transitory memory, the touch screen, and the vibrator:

158. See discussion of [30.d] (Ground 1C).

[62.f] displaying a first object and at least one other object;

159. See discussion of [30.e] (Ground 1C).

[62.g] detecting a gesture on the touch screen on the first object;

160. See discussion of [30.f] (Ground 1C).

[62.h] in response to a detection of at least a portion of the gesture, performing a scale operation, based on a change in a duration of the gesture being

detected on the touch screen on the first object, on at least the at least one other object,

161. See discussion of [30.g] (Ground 1C).

[62.i] where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected on the touch screen on the first object; and

162. See discussion of [30.h] (Ground 1C).

[62.j] subsequent to the performance of the scale operation, vibrating the apparatus and performing an action.

163. See discussion of [30.i] (Ground 1C).

E. Ground 2A: Claims 1, 27, and 32 are obvious over Kim in view of Bowens

1. Bowens

164. Bowens is directed to a “state machine gesture recognition algorithm for interpreting streams of coordinates received from a touch sensor.” APPLE-1005, Abstract. “A touch,” it explains, “is usually only output as a laid touch if certain criteria are satisfied,” and “[i]f a touch of a duration longer than a threshold duration is sensed in the touch state, then control flow passes to a press state module, wherein the press state is for handling longer touches.” *Id.*, [0023].

165. Bowens describes various types of user actions and how each are identified. APPLE-1005, [0055]-[0077], FIGS. 1-3. For example, a “tap event is generated when the user makes a tap gesture” (i.e., “when the user quickly touches and releases the touch surface”). *Id.*, [0056], [0065]. A tap “could be used, for

example, to active a hyperlink on a displayed web page.” *Id.*, [0056]. “A double tap happens when the user quickly touches and releases the touch surface twice in quick succession,” and can be “used, for example, to select a word in a displayed document.” *Id.*, [0057].

166. “A press event is generated when the user touches the touch surface and leaves their finger stationary.” *Id.*, [0067]. “A short press event is generated after a press event, when the user has continued to leave their finger stationary for a period,” and a “long press event is generated after a short press event, when the user has continued to leave their finger stationary for a period.” *Id.*, [0068]-[0069]. “A repeat press event is generated after a long press event, when the user has continued to leave their finger stationary for a period,” and can be “generated at regular intervals” if the user “leave[s] their finger stationary.” *Id.*, [0070].

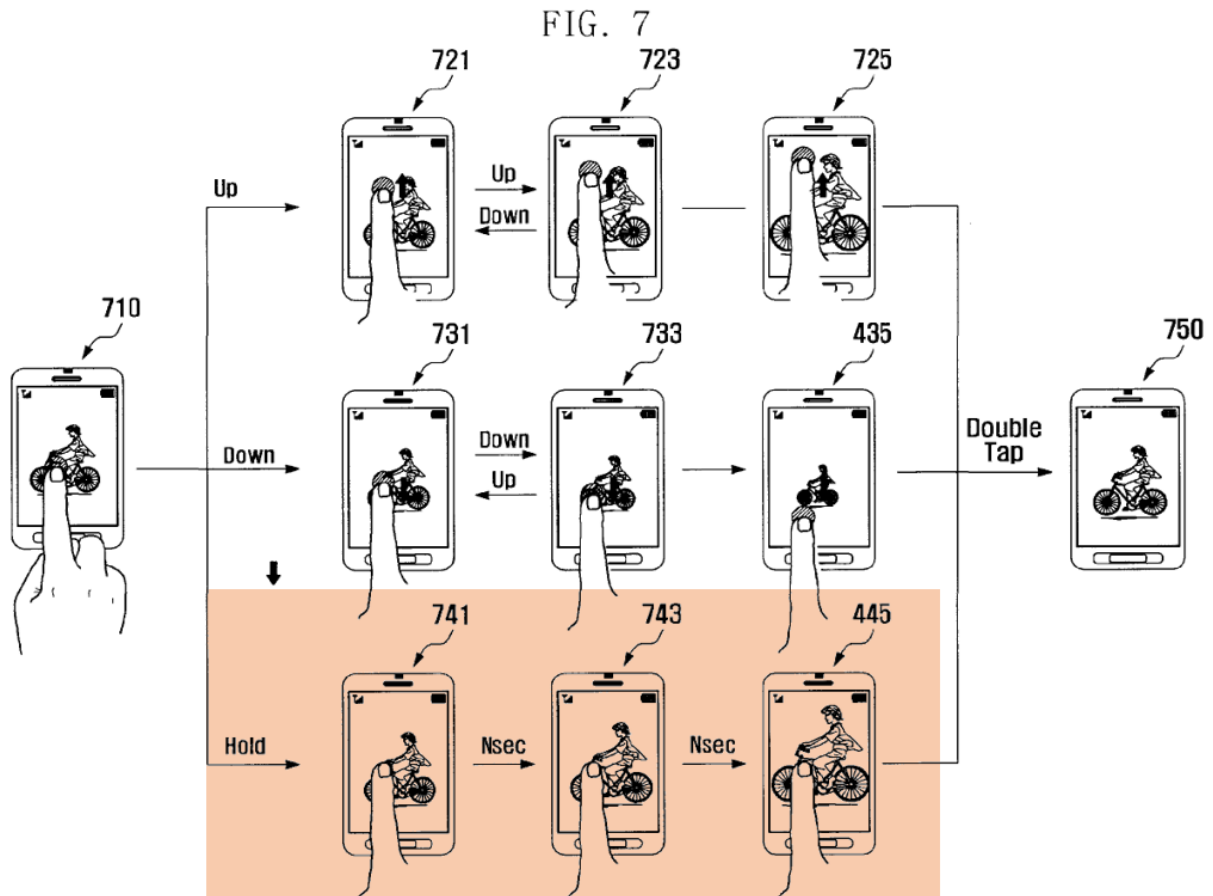
2. The Kim-Bowens Combination

167. Kim describes a “mobile device” with “a touch sensitive surface such as a touch screen” that provides a “touch-based zooming control method.” APPLE-1004, Abstract. With this method, “[w]hile the target data is displayed, the mobile device activates a zoom mode in response to a given gesture or input event,” which causes “a zooming function such as a zoom-in and a zoom-out in response to the received gesture input.” *Id.* Kim explains that “a zoom mode can

be activated when a user invokes a *long press* at a certain point,” and can “perform a zooming action” in two ways. APPLE-1004, [0139] (emphasis added).

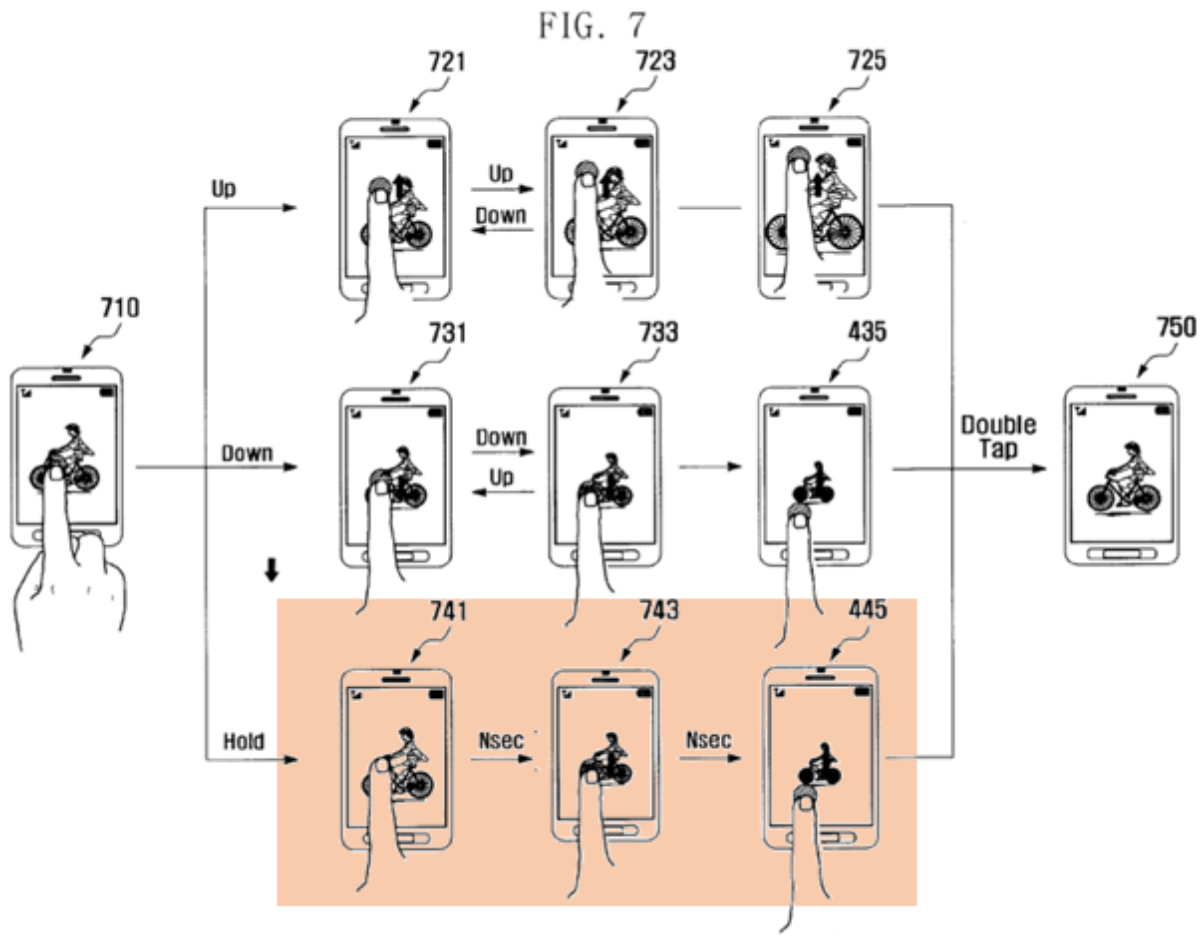
168. With respect to Figure 7 (below), Kim describes the two ways (moving and holding): “a user can manipulate the mobile device to perform a continuous zooming function (namely, a zoom-in or a zoom-out) through an appropriate gesture such as a move event or a hold event.” APPLE-1004, [0095], FIG. 7; *see id.*, [0096]-[0107], [0139]-[0140]. “[R]eference numbers 721, 723 and 725 show a continuous zoom-in action depending on a move [up] event,” and “reference number 731, 733 and 735 show a continuous zoom-out action depending on a move [down] event.” *Id.*, [0096], [0100]. Referencing numbers 741, 743, and 745⁷, Kim “show a continuous zoom-in action depending on a hold event.” APPLE-1004, [0104], [0140]. A user “can invoke the hold event at a selected point through a touch input such as a tap and hold.” *Id.*

⁷ Reference number “445” in Figure 7 appears to be a typo.



APPLE-1004, FIG. 7 (highlighted)

169. Kim explains that, while Figure 7 shows “the hold event is assigned to a continuous zoom-in function,” in other scenarios, “the hold event may alternatively assigned to a continuous zoom-out function.” APPLE-1004, [0107]. This alternative is illustrated below in modified Figure 7, which shows the object (person on a bicycle getting smaller as the hold event persists).



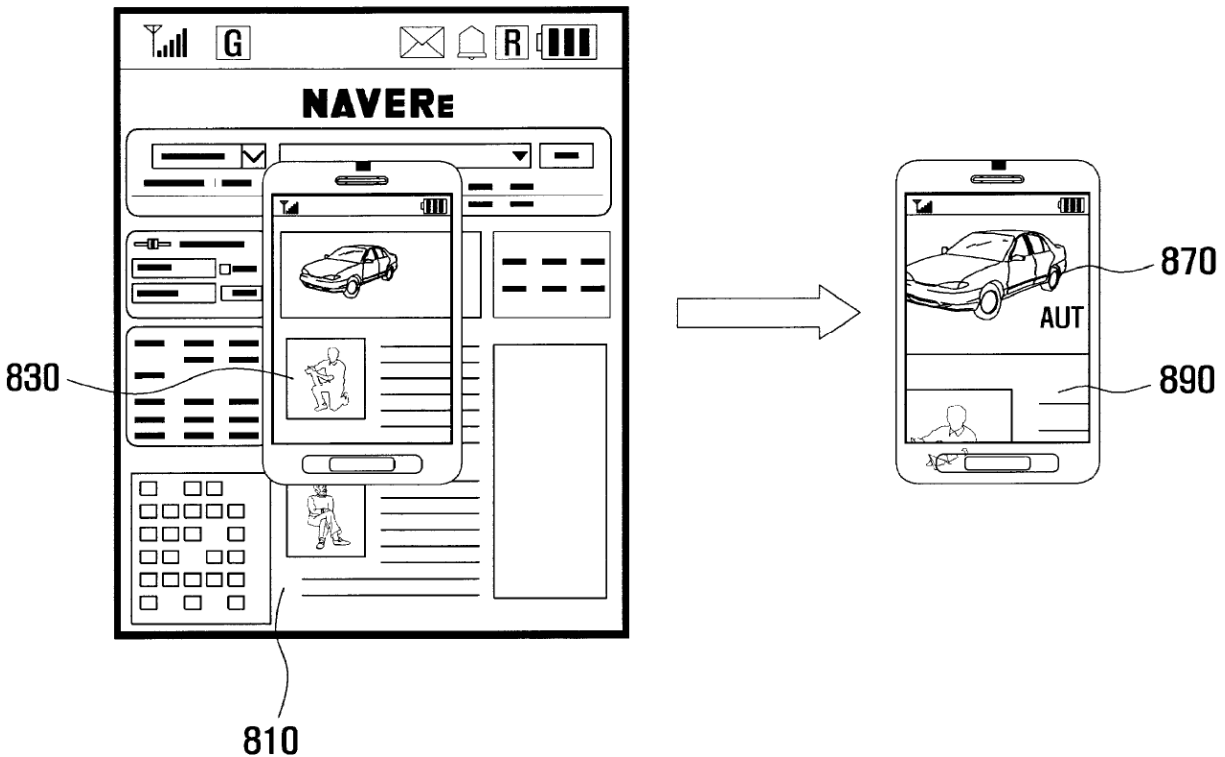
APPLE-1004, FIG. 7 (“hold” scenario modified)

170. In another example shown in Figure 8 (below), Kim shows “an example of a zooming action associated with web display data in a web browser environment.” APPLE-1004, [0113]. Kim explains, “Normally web display data can contain a variety of link information about any of a transfer to other web page and a file download,” and provides for “such display target data, a process of a zoom-in or zoom-out action.” *Id.* In this example, Kim notes that “viewport 830 contains a particular link assigned to certain text ‘AUTO’ with coordinate (x1,

y1),” and the device provides a technique to “perform a conversion operation for the coordinate of ‘AUTO’ with link information whenever one of enlarging and reducing the display target data in response to a user’s request.” *Id.*, [0114]. When a user requests “one of enlarging and reducing the display target data,” the “mobile device performs a processing of a continuous zooming action by using the bitmap image and offers the display target data one of and[sic] enlarged and reduced.” *Id.*, [0115]. Notably, while Figure 8 shows enlarging (e.g., zooming in), Kim also teaches reducing the size of the target data (e.g., zooming out). APPLE-1004, [0113]-[0115]. Thus, Kim teaches a device that allows for performing a continuous zoom-in (enlarging) or zoom-out (reducing) function responsive to a user request (e.g., either a tap and move up or move down, or a tap and hold (long press)), where the zooming function can be performed on “web display data in a web browser environment.” APPLE-1004, [0094]-[0115], FIGS. 7-8.

Additionally, Kim teaches that after a zoom action, “a user can invoke a double tap event on the screen,” which causes “the mobile device [to] display[] target data with a given basic magnification value, as indicated by reference number 750.” APPLE-1004, [0099], [0103], [0106], FIG. 7.

FIG. 8



APPLE-1004, FIG. 8

171. While Kim provides these teachings relating to a zoom function, Kim leaves to well-informed POSITAs various specific implementation details pertaining to how the system identifies and differentiates the various touches and gestures. *See* APPLE-1004, [0053] (“If detecting a user gesture, namely, if receiving an input event from a user through the touch screen, the mobile device determines the type of an input event”). As such, a POSITA would have naturally looked to and reasonably expected to succeed in integrating the disclosed detail found in prior art related to touch gestures, such as Bowens, which includes, for example, implementation details and suggestions on how to distinguish between

different types of touch gestures and their corresponding operations, as was well known by the Critical Date.

172. As discussed above, Bowens is directed to a “state machine gesture recognition algorithm for interpreting streams of coordinates received from a touch sensor.” APPLE-1005, Abstract, FIGS. 1-3. Like Kim, Bowens describes “events” that are generated in response to detected gestures. *See, e.g.*, APPLE-1004, [0035] (“This input event can be invoked through a double tap input occurring at a specific point on a touch screen”); APPLE-1005, [0065] (“A tap event is generated when the user makes a tap gesture.”), [0066]-[0069].

173. Significantly, Bowens describes various types of user gestures are identified and distinguished from one another. APPLE-1005, [0055]-[0077]. For example, a “tap event is generated when the user makes a tap gesture” (i.e., “when the user quickly touches and releases the touch surface”) and could be used, for example, to active a hyperlink on a displayed web page.” *Id.*, [0056], [0065]. In more detail, Bowens teaches that the “touched state occurs when a user has touched the touch surface,” and “[w]hen the Touched state is entered a timeout is started, i.e. a timer is set running which expires after a preset time.” *Id.*, [0080]. “This is used to distinguish between short duration gestures such as flicks and taps, and long duration gestures such as a press.” *Id.*

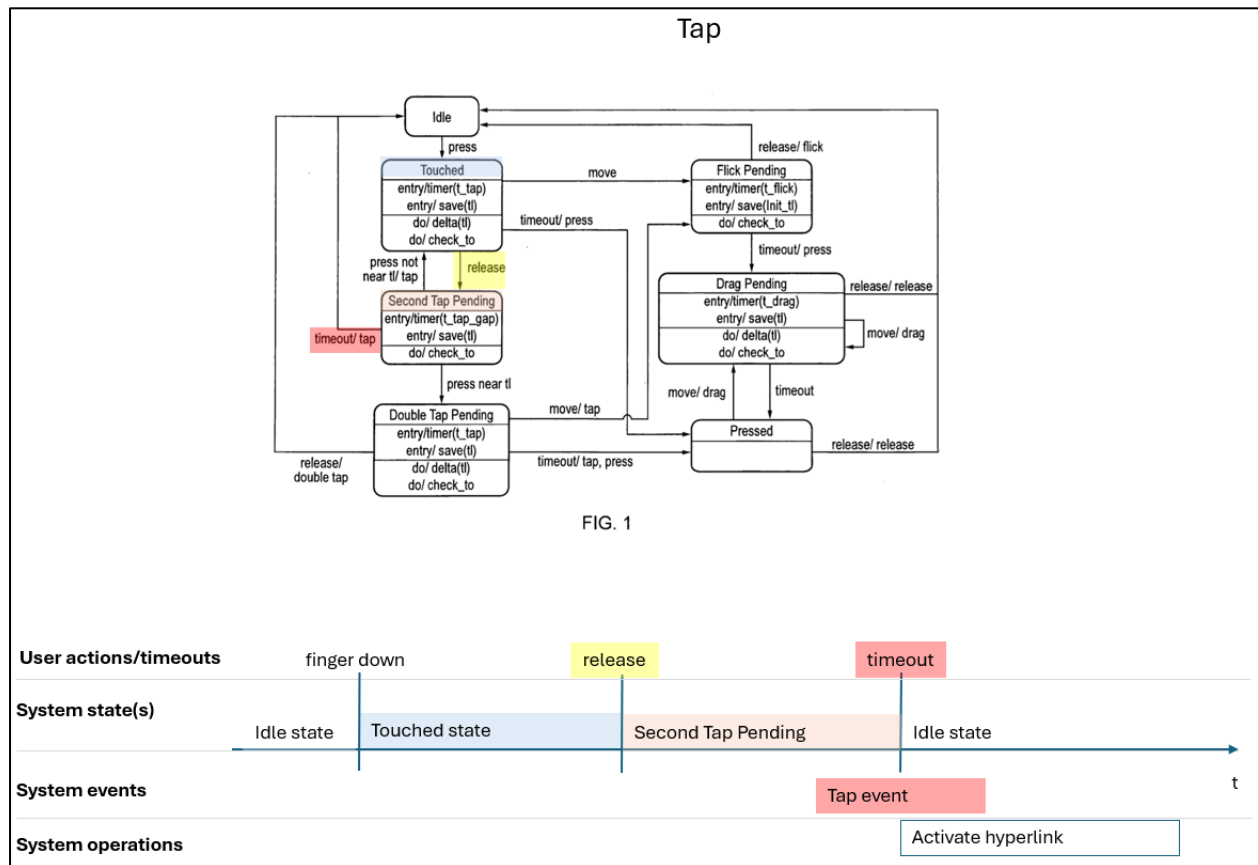
174. Bowens explains that, in this Touched state, if the user releases the touch, “the state machine enters the Second Tap Pending state,” after which “[s]ubsequent actions will determine if a tap, double tap, or other tap” is generated. *Id.* The “Second Tap Pending state occurs after a user has tapped the touch surface” and causes a timeout to start, which “is used to decide if the user has made a tap gesture.” *Id.* If the timeout expires, “a tap event is generated.” *Id.*, [0081]. However, if “the user presses the touch surface” during the Second Tap Pending state and at a location within threshold distance of the location of the first touch, “the state machine enters the Double Tap pending state,” which is later confirmed as a double tap if the user releases his touch within a timeout parameter. *Id.*, [0082].

175. Bowens additionally explains that, “[i]f the [Touched state] timeout expires, the state machine undergoes a transition by generating a press event and then enters the Pressed state.” APPLE-1005, [0080]. After entering the Touched state, the “initial touch location is stored.” *Id.* Bowens explains that “[i]f a touch of a duration longer than a threshold duration is sensed in the touch state, then control flow passes to a press state module, wherein the press state is for handling longer touches.” APPLE-1005, [0023]; *see also id.*, [0029], [0080]. Notably, like Kim, Bowens contemplates a “long press”: A “Long Pending state is for when a user has pressed the touch surface and has been stationary for some time,” at which

point “a timeout is started,” which is “used to determine when to generate a long press event.” APPLE-1005, [0085]-[0098], FIG. 3. When “the timeout expires, a long press event is generated.” *Id.*

176. In view of Bowens’s teachings with respect to detecting and distinguishing between different types of user inputs, a POSITA would have found it obvious to implement Kim’s device with methods for how to recognize different gestures (e.g., distinguishing between a tap, a double tap, and a long press) and with well-known tap-related functions, as Bowens teaches and suggests. Specifically, a POSITA would have implemented the Kim-Bowens device (“KBD”) such that the device uses a gesture recognition algorithm for addressing received user inputs and for well-known tap-related functionality, as Bowens teaches and suggests. In one example, in the KBD, after the KBD receives a touch input on the touch screen, the KBD first enters a Touched state, storing the location of the touch and starting a timer (t_{tap}). APPLE-1005, [0080] (explaining that the timer is “used to distinguish between short duration gestures such as flicks and taps, and long duration gestures such as a press”), [0023], FIG. 2. If the user releases his/her touch during the Touched state, the device enters a “Second Tap Pending state,” and a timer ($t_{\text{tap_gap}}$) starts. *Id.*, [0081], FIG. 2. As in Bowens, in the KBD, the “Second Tap Pending state occurs after a user has tapped the touch surface” and causes a timeout to start, which “is used to decide if the user has made

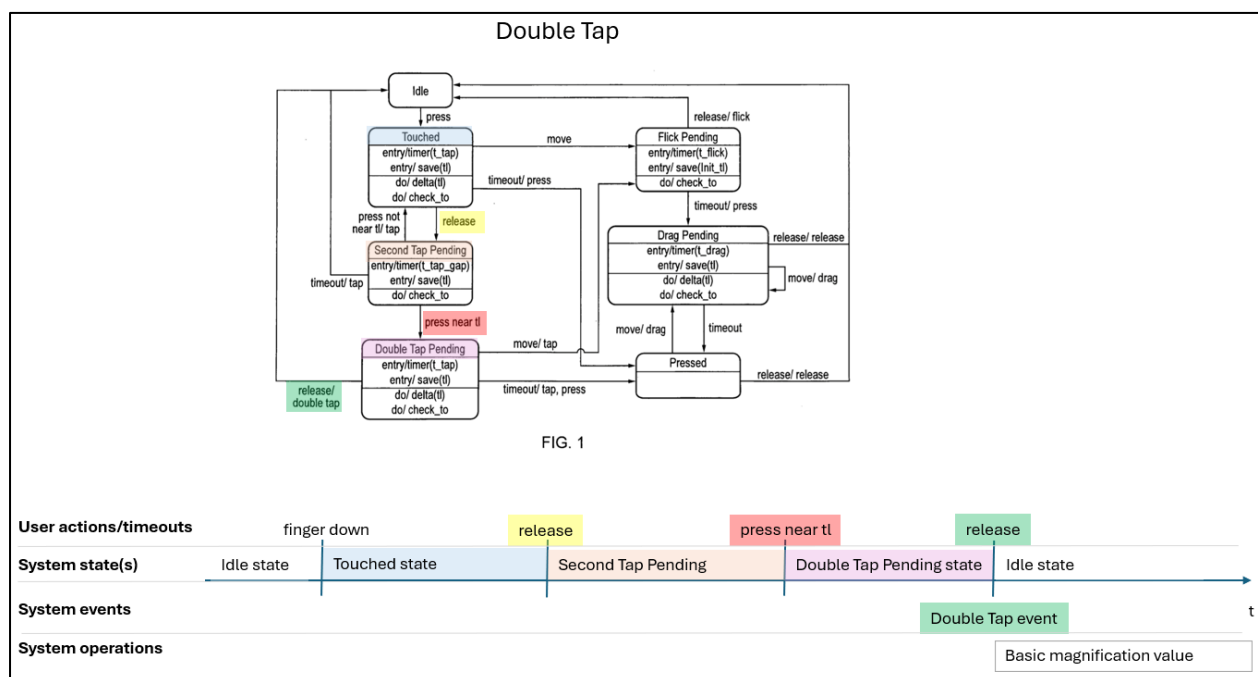
a tap gesture.” *Id.* If the timeout expires, “a tap event is generated.” *Id.* If that timeout expires, “a tap event is generated” and the gesture is then identified as a tap, which, as Bowens teaches, “activates a hyperlink on a displayed web page.” *Id.*, [0056], [0081]. An illustration of a “tap event” in the KBD is provided below using Bowens’s Figure 1 flow chart:



APPLE-1005, FIG. 1 (highlighted; annotated)

177. The KBD would further use a method for identifying a double tap, as Bowens teaches and suggests. Specifically, during the Second Tap Pending state, if another touch is detected at a location near to the first touch prior to the expiry of the timeout, then the state is changed to “Double Tap Pending.” APPLE-1005,

[0081]-[0082]. “The distance of the second touch from the first is calculated. If it is less than a specified value, the state machine enters the Double Tap Pending state, as the user may be doing a double tap.” *Id.*, [0081]-[0082], FIG. 2. In the KBD, if a double tap is detected (“the state machine generates a double tap event and enters the Idle state,” per Bowens), the “device displays target data with a given basic magnification value.” APPLE-1004, [0106]; APPLE-1005, [0082]. An illustration of a “double tap event” in the KBD is provided below using Bowens’s Figure 1 flow chart:

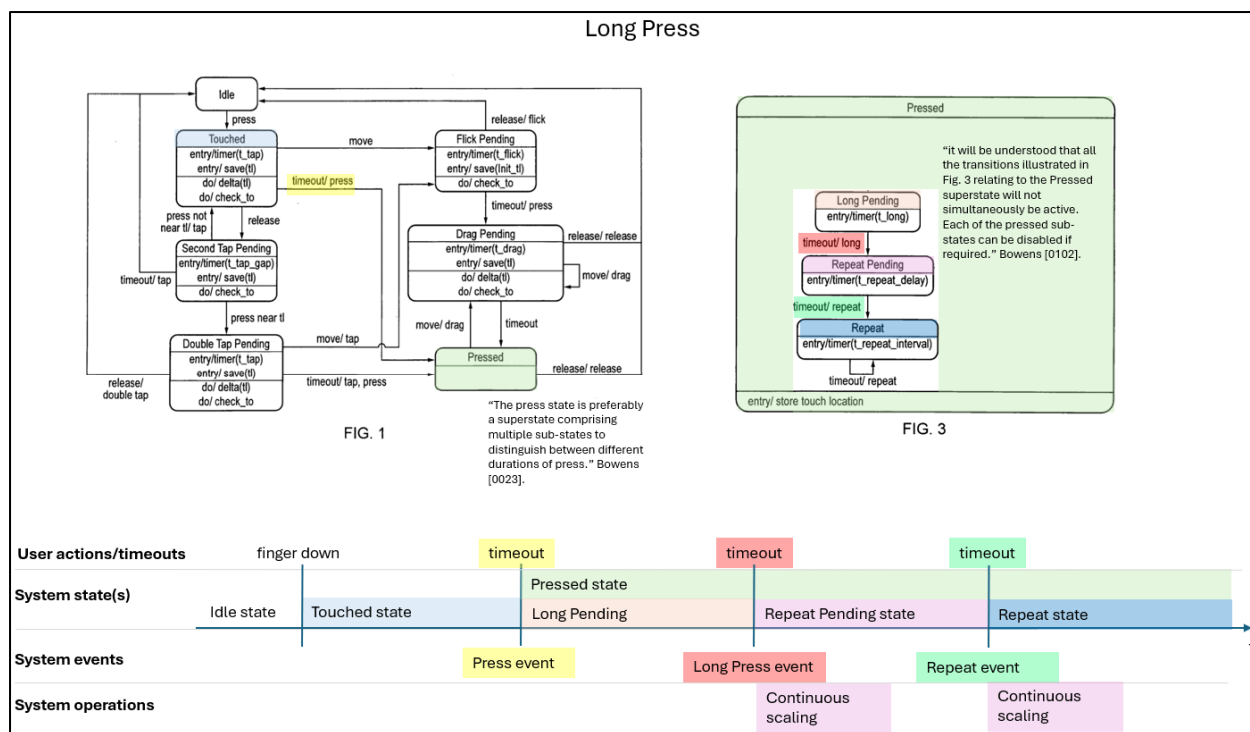


APPLE-1005, FIG. 1 (highlighted; annotated)

178. If the touch persists for a longer period of time (from the Touched or Double Tap Pending states), “then control flow passes to a press state module, wherein the press state is for handling longer touches.” APPLE-1005, [0023],

[0080] (“If the timeout expires, the state machine undergoes a transition by generating a press event and then enters the Pressed state.”), FIGS. 2-3. When a “user has pressed the” touch screen of the device “and been stationary for some time,” the device is in the “Long Pending state,” which starts a timer (t_long) on entry. APPLE-1005, [0098], FIG. 3. “If the timeout expires, a long press event is generated and the state machine enters the Repeat Pending or Press Complete states,” as Bowens describes, which a POSITA would have understood as corresponding to Kim’s “tap and hold” or “long press,” which causes the KBD to *invoke a continuous zoom-in or zoom-out function*. APPLE-1004, [0104]-[0107], [0139]-[0140], FIG. 7; APPLE-1005, [0098].

179. An illustration of a “long press event” in the KBD is provided below using Bowens’s Figure 1 and Figure 3 flow charts:



APPLE-1005, FIGS. 1 and 3, [0023], [0102] (highlighted; annotated)

180. A POSITA would have understood that implementing a device (per Kim) with tap-related operation (activate a hyperlink on a displayed web page), per Bowens, would offer distinct functions associated with a tap, double tap, and a long press, respectively. APPLE-1005, [0056]. Indeed, Kim offers an exemplary scenario where “a zooming action associated with web display data in a web browser” is provided. APPLE-1004, [0113]. It would have been obvious to a POSITA that the device would respond with different operations depending on the type of input received.

181. A POSITA would have further understood that implementing the device with “gesture recognition algorithm for interpreting streams of coordinates

received from a touch sensor,” per Bowens, would provide a clear method for distinguishing between the different types of touch gestures Kim describes.

APPLE-1005, Abstract. Specifically, a POSITA would have understood that implementing Kim’s device with the various “states” (e.g., Touched state, Second Tap Pending state, Idle state, Pressed state), and setting timers when entering each state, would provide the implementation details for Kim’s teachings of distinguishing a tap from a double tap or long press. APPLE-1004, [0139], APPLE-1005, [0080]. A POSITA would have understood that applying Bowens’s teachings and using timers/timeouts (and threshold durations) for distinguishing between taps, double taps, and long presses in the device would have led to predictable results without significantly altering or hindering the functions performed by the device, particularly because Kim already contemplates receiving different types of inputs (tap, long press, double tap). APPLE-1004, [0042], [0053], [0103]-[0104], [0106]. Kim further contemplates determining the particular type of input a gesture is. APPLE-1004, [0042] (“the mobile device can determine whether an interrupt is a long press input or a double tap input...”), [0049], [0053] (“the mobile device determines whether an input event is one of a move event [], a hold event [], and a double tap event”), [0072]. A POSITA would have been motivated to implement the device (per Kim) with Bowens’s teachings in the KBD to achieve the predictable results offered by Bowens’s description of

algorithms to detect gestures, related gesture events, and how such gestures and events are determined. APPLE-1005, [0055]-[0077]. Because recognizing predetermined touch gestures was known, and Kim and Bowens both describe the use of touchscreen devices and predetermined criteria for registering touches on the devices, a POSITA would have reasonably understood the combination of Kim and Bowens to be successful with no unexpected results. APPLE-1004, [0003], [0011], [0030], [0039], [0041], [0094]-[0107], [0122]-[0137], FIGS. 5-9; APPLE-1005, Abstract, [0018], [0039], [0055]-[0078], [0080]-[0101], [0141].

182. As explained above, combining Kim and Bowens would have been obvious at least because such a combination would have merely involved (1) combining prior art elements according to known methods to yield predictable results and (2) the use of known technique to improve similar devices (methods, or products) in the same way. Furthermore, the elements of the resulting combination would each perform functions they had been known to perform prior to the combination—the device would perform the same functions to zoom-out responsive to a long press, and receive different types of input gestures, as taught by both Kim and Bowens, and would leverage Bowens’ teachings regarding the implementation details of how to distinguish between the different gestures to ultimately perform the respective operation. Accordingly, a POSITA would have naturally expected success when incorporating Bowens’s teachings into Kim. *Id.*

3. Claim 1

[1.pre] An apparatus, comprising:

183. See discussion of [1.pre] (Ground 1A).

[1.a] at least one non-transitory memory storing instructions and a plurality of applications;

184. See discussion of [1.a] (Ground 1A).

[1.b] a touch screen; and

185. See discussion of [1.b] (Ground 1A).

[1.c] 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 the instructions to:

186. See discussion of [1.c] (Ground 1A).

[1.d] display, via the touch screen, an object and at least a portion of an interface;

187. See discussion of [1.d] (Ground 1A).

[1.e] detect a gesture on the touch screen on the object;

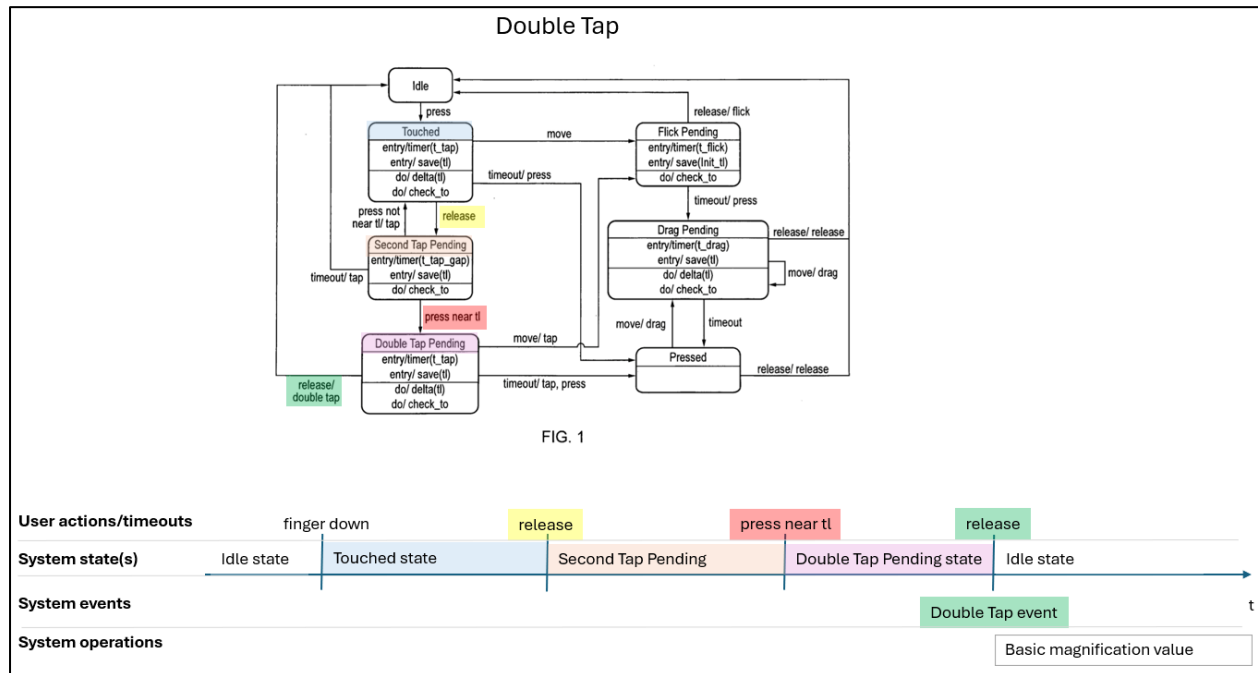
188. See discussion of [1.e] (Ground 1A).

189. Moreover, like Kim, Bowens teaches a device that recognizes and generates events for different gesture inputs, such as a tap, a press, and a long press. APPLE-1005, [0055]-[0077].

[1.f] when a magnitude of a duration of the gesture being detected on the touch screen on the object is less than a first magnitude threshold, perform an operation;

190. The KBD renders [1.f] obvious. APPLE-1004, FIGS. 5-8. As discussed above, the KBD includes various system states and corresponding operations. §VII.E.2. In one example, the KBD can generate a double tap event. APPLE-1005, [0081]-[0082]. As illustrated below, a double tap event is detected based on a duration of the touch, which is less than a first magnitude threshold (e.g., the timeout for the transition from Touched to Pressed). *Id.* In particular, a POSITA would have understood that the duration of each of the taps of the double tap gesture must be less than a timeout because, if either timeout expires, the gesture cannot be a double tap. For the first tap, the release must occur before the “timeout/ press” transition from the Touched state to the Pressed state. APPLE-1005, FIG.1. For the second tap, the release must occur before the “timeout/ tap, press” transition from the Double Tap Pending state to the Pressed state. *Id.* The duration of each of the taps of the double tap gesture is “a duration of the gesture.” A POSITA would have understood that gesture recognition encompasses identification of multiple components of touch data. Consistent with this understanding, the specification of the ’114 patent states that “gesture input refers to the collection of inputs, flags, and signals by which a gesture is identified and parameterized,” that “[p]ossible gesture inputs may include, but are not limited to,

touch event attributes (both initial and over time),” and that “[s]ome gestures may be able to be identified solely from initial touch event attributes. Other gestures may only be identified after the gesture has been performed for some period of time.” APPLE-1001, 58:13-15, 58:19-20, 58:27-30. In the KBD, when a double tap event is generated, the “device displays target data with a given basic magnification value,” thus performing another operation. APPLE-1004, [0106]; APPLE-1005, [0082].



APPLE-1005, FIG. 1 (highlighted; annotated)

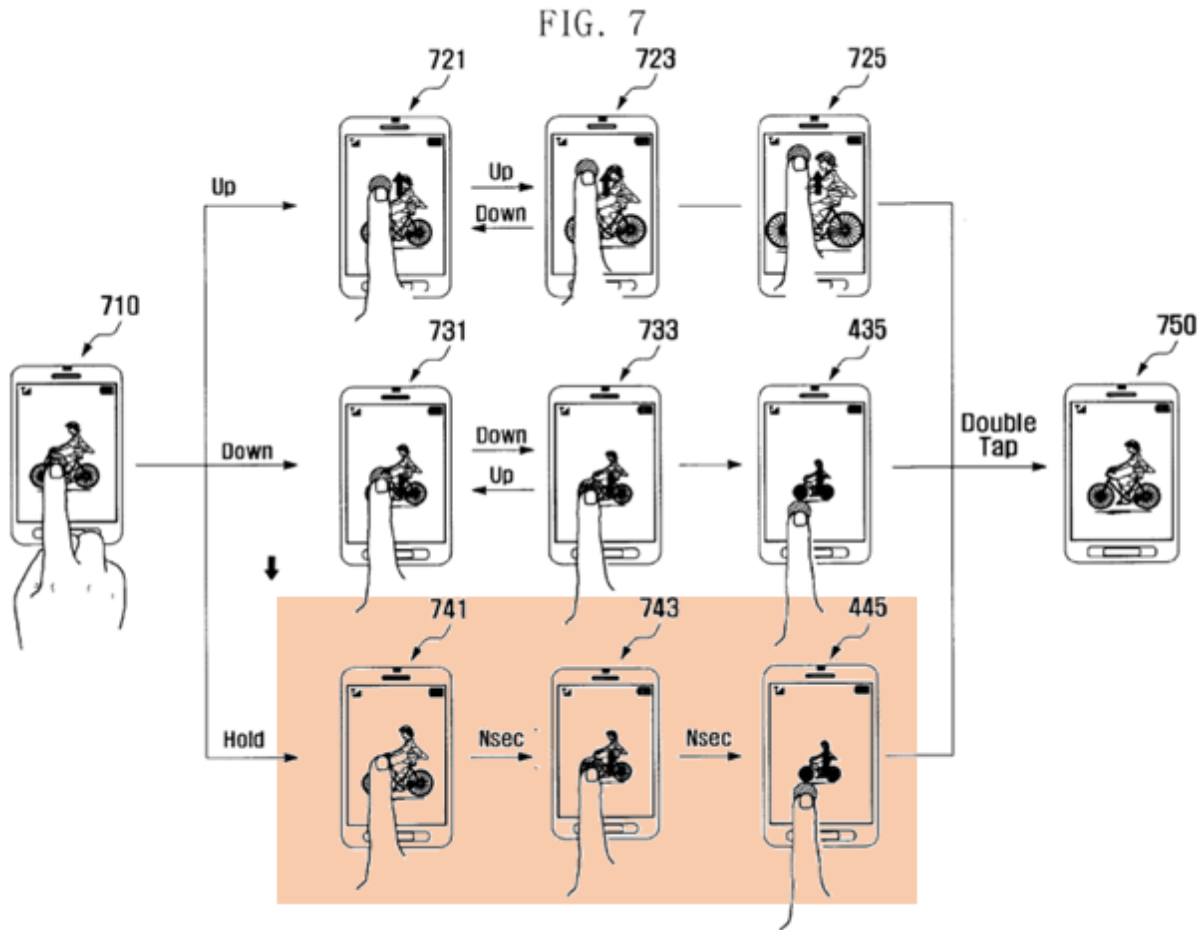
[1.g] when the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than the first magnitude threshold, perform a scaling in connection with at least the object,

191. The KBD renders [1.g] obvious. APPLE-1004, FIGS. 5-8. As discussed above, if contact with the touch screen persists beyond the Touched state

to a Pressed state and then to a Long Pending state, a timer (t_{long}) starts.

§VII.E.2; APPLE-1005, [0098], FIG. 3. “If the timeout expires, a long press event is generated and the state machine enters the Repeat Pending or Press Complete states,” as Bowens describes, which a POSITA would have understood as corresponding to Kim’s “tap and hold” or “long press,” which causes the KBD to *invoke a continuous zoom-in or zoom-out function*. APPLE-1004, [0104]-[0107], [0139]-[0140], FIG. 7; APPLE-1005, [0098]. Bowens teaches that the “long press event” is generated when the duration of the touch is longer, for example, than a tap (or double tap) (e.g., greater than the first magnitude threshold). APPLE-1005, [0023] (“If a touch of a duration longer than a threshold duration is sensed in the touch state, then control flow passes to a press state module, wherein the press state is for handling longer touches.”), FIG. 3. While Figure 7 shows the object (person on bicycle) at reference numbers 741, 743, and 745 increasing in size (e.g., zoom in), Kim explains that “the hold event may be alternatively assigned to a *continuous zoom-out function*” such that the tap and hold input causes the object (person on bicycle) to decrease in size (e.g., zoom out). *Id.*, [0107] (emphasis added), FIG. 7. Modified Figure 7 (below) illustrates Kim’s alternative assignment. A POSITA would have understood that this same functionality can also be applied to other content, including, for example, the

content discussed in Figure 8 that includes a displayed hyperlink. APPLE-1004, [0114]-[0115], [0145], FIG. 8.



APPLE-1004, FIG. 7 (“hold” scenario modified)

192. Thus, when the magnitude of the duration of the gesture on the touch screen on the object is greater than the first magnitude threshold (e.g., the timeout for the transition from Touched to Pressed), the KBD performs a scaling of the object (e.g., hyperlink, image, etc.).

[1.h] where the scaling includes a reduction in a size such that the size is inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the object; and

193. The KBD renders [1.h] obvious. A POSITA would have understood that a “continuous zoom-out function” on an object, such as the bike and bicyclist of Kim’s Figure 7 or the hyperlink of Figure 8, is the equivalent of scale operation that includes a reduction in size of the object. Kim explains that, when the hold event is assigned to a zoom-in process, “the mobile device displays an enlarged display target data.” APPLE-1004, [0106]. “In such a zoom-in process, the mobile device continuously offers real-time, scaled-up bitmap image on the screen in response to the duration of the hold event.” *Id.*, [0105]. A POSITA would have understood that, when the hold event is assigned to a zoom-out process, the mobile device would instead display a scaled-down bitmap image on the screen in response to the duration of the hold event. APPLE-1004, [0031] (“a gesture refers to a touch-based user’s input motion which travels in a given pattern assigned to a zoom-in or zoom-out function in a zoom mode”), [0106] (“the hold event may be alternatively assigned to a continuous zoom-out function”).

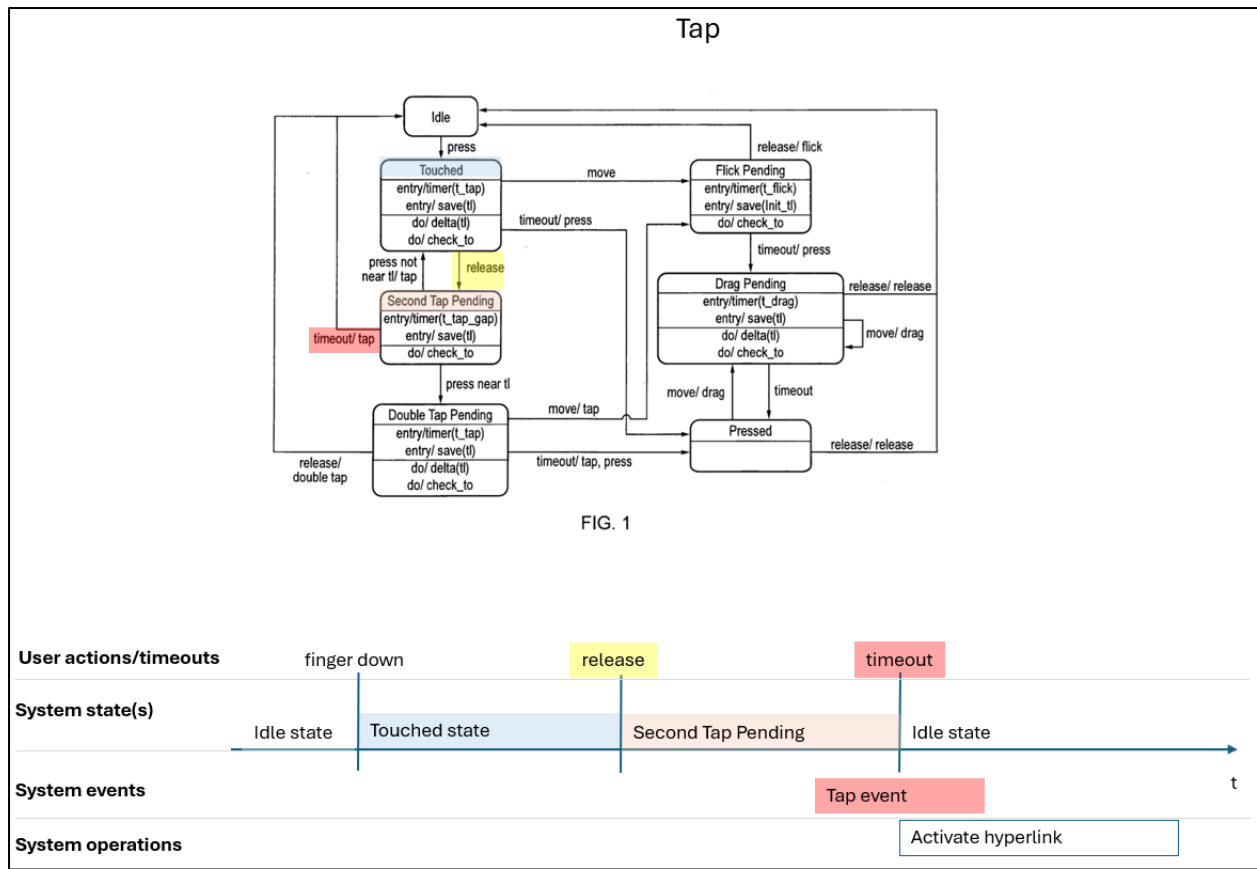
194. Additionally, Kim teaches that the zoom-out is “continuous”: in a zoom-out process, “the mobile device continuously offers a real-time, scaled-[down] bitmap image on the screen in response to the duration of the hold event.” APPLE-1004, [0105]. Thus, the longer the duration of the gesture (press and

hold), the more the object is zoomed-out, or the more the size of the object is reduced. *Id.* In this way, the reduction in a size is inversely related to the magnitude of the duration of the gesture (the larger the duration of the gesture, the smaller the size). *Id.*

[1.i] when the magnitude of the duration of the gesture being detected on the touch screen on the object is greater than a second magnitude threshold, perform another operation.

195. The KBD renders [1.i] obvious. As discussed above with respect to [1.e], the detected gesture may be received on the touchscreen on an object (e.g., a hyperlink). As discussed above, “after a user has tapped the touch surface” and “the user releases their touch” (highlighted yellow below), a timeout starts in the KBD device, which “is used to decide if the user has made a tap gesture.”

§VII.E.2; APPLE-1005, [0065], [0080]-[0081]. If that timeout expires, “a tap event is generated” and the gesture is then identified as a tap, which, as Bowens teaches, “activates a hyperlink on a displayed web page.” APPLE-1005, [0056], [0081]. A POSITA would have understood that if the gesture is determined to be a tap gesture (which ceased at the time labeled “release,” highlighted yellow below), then the magnitude of the duration of the gesture on the touch screen is greater than a second magnitude (the “timeout/tap” transition from SecondTapPending to Idle). In this scenario, the KBD performs another operation (activating the hyperlink). APPLE-1005, [0056], [0081].



APPLE-1005, FIG. 1 (highlighted; annotated)

4. Claim 27

[27.pre] *An apparatus, comprising:*

196. *See discussion of [1.pre] (Ground 1A).*

[27.a] *at least one non-transitory memory storing instructions and a plurality of applications;*

197. *See discussion of [1.a] (Ground 1A).*

[27.b] *a touch screen; and*

198. *See discussion of [1.b] (Ground 1A).*

[27.c] 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 the instructions to cause the apparatus to:

199. See discussion of [1.c] (Ground 1A).

[27.d] display an object on at least a portion of an interface;

200. See discussion of [1.d] (Ground 1A).

[27.e] detect a gesture via the touch screen on the object;

201. See discussion of [1.e] (Ground 1A).

[27.f] perform a first function in a first touch state;

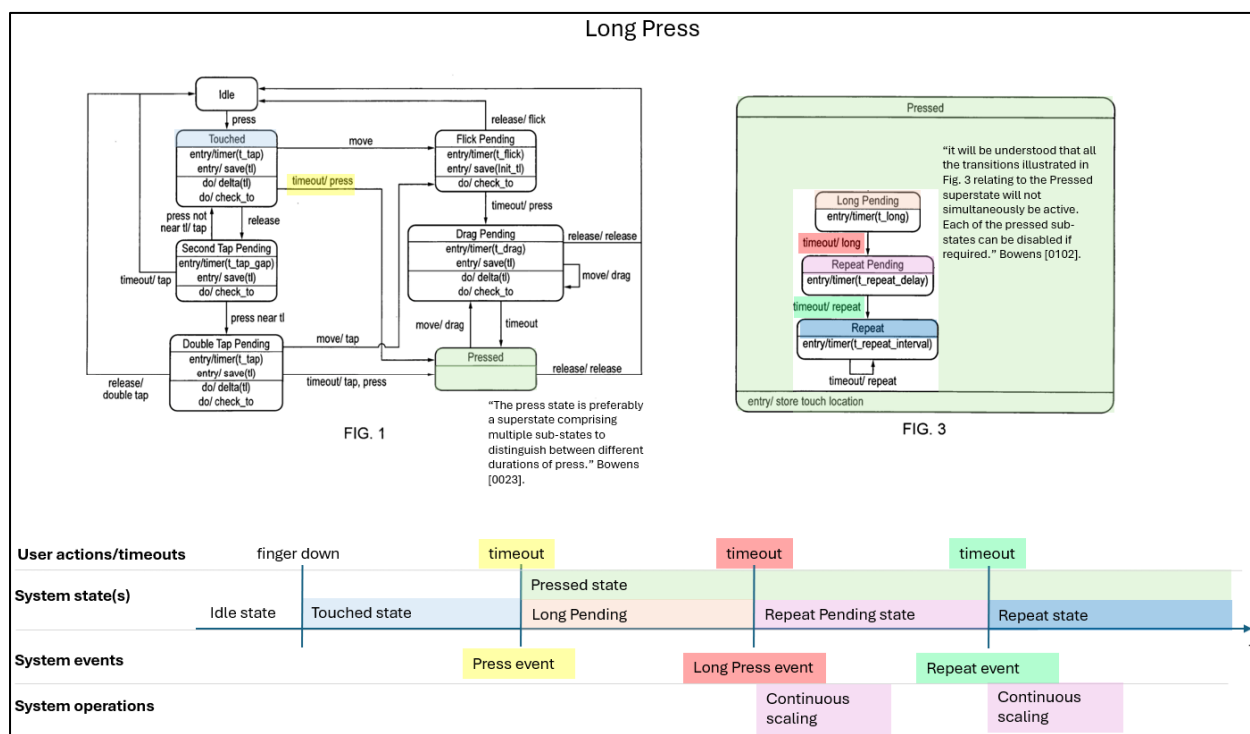
202. The KBD renders [27.f] obvious. As discussed above, the KBD is in a first touch state (“Second Tap Pending”) “after a user has tapped the touch surface,” which causes the device to generate a “tap event.” §VII.E.2; APPLE-1005, [0065], [0081]. In the KBD, when a tap event occurs in a “a web browser environment,” a “hyperlink on a displayed web page” is activated. APPLE-1004, [0113], FIG. 8; APPLE-1005, [0056]. Thus, in the Second Tap Pending state, the function of activating a hyperlink is performed. *Id.*

[27.g] perform a second function in a second touch state; and

203. The KBD renders [27.g] obvious. As discussed above, in the KBD, a “Double Tap Pending state occurs when a user has tapped the touch surface and pressed it again.” §VII.E.2; APPLE-1005, [0082]. When the KBD is in a “Double Tap Pending state” and “the user releases their touch [from the second touch], the state machine generates a double tap event and enters the Idle state.” *Id.* In this

second touch state (“Idle state”) following the double tap detection, the KBD performs a second function (e.g., “displays target data with a given basic magnification value”). APPLE-1004, [0106], FIG. 7; APPLE- 1005, [0057], [0082].

204. Alternatively, when the KBD is in a “Repeat Pending state” (second touch state) following a “Long Press event,” the device performs continuous scaling (second function), as discussed in more detail with respect to [27.h]. See §VII.E.2, discussion of [27.h].

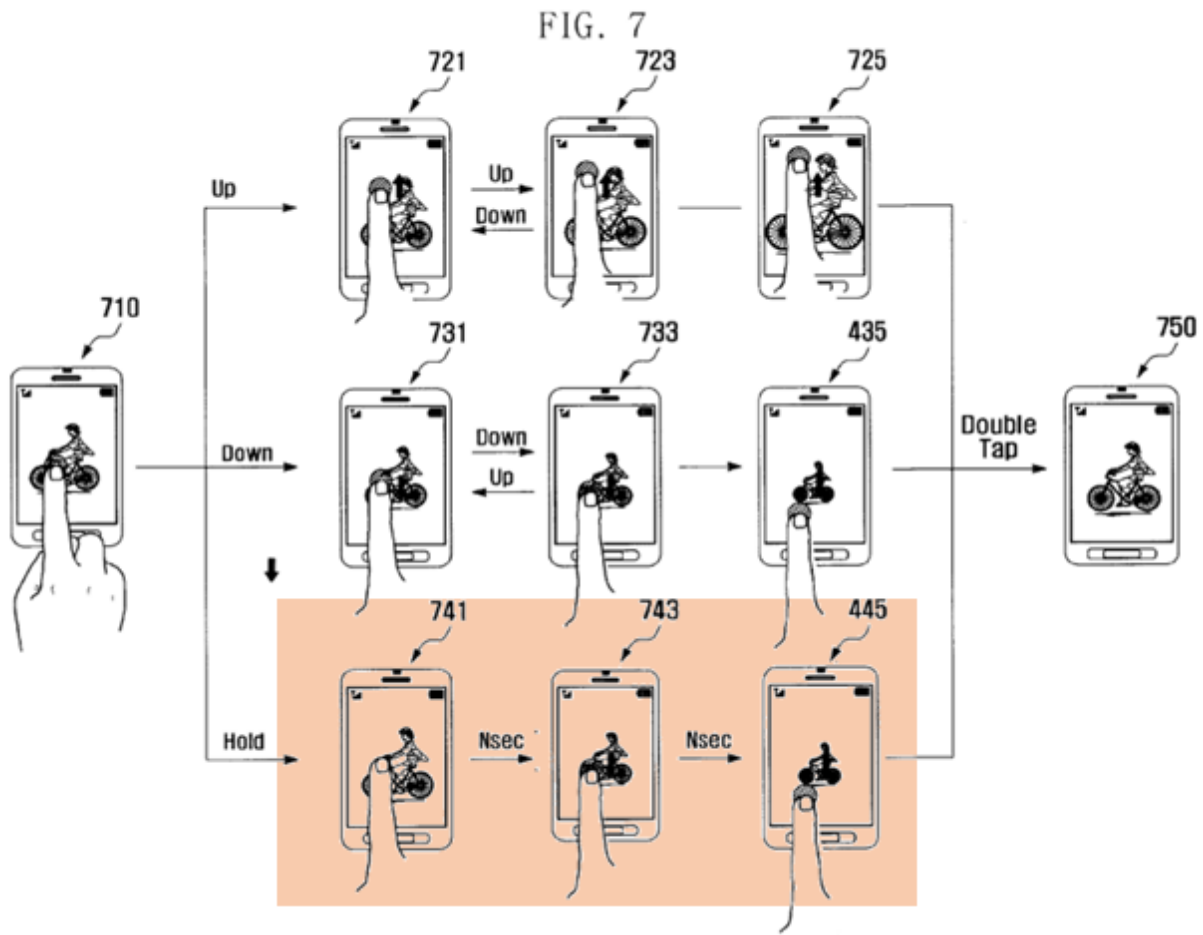


APPLE-1005, FIGS. 1 and 3; [0023], [0102] (highlighted; annotated)

[27.h] during detection of at least a portion of the gesture and based on a change in a duration of the gesture being detected via the touch screen on the object, perform a scale operation in connection with the at least portion of the interface, where the scale operation includes a reduction in a size that

is inversely related to the duration of the gesture being detected via the touch screen on the object.

205. The KBD renders [27.h] obvious. As discussed above, if contact with the touch screen persists beyond the Touched state to a Pressed state and then to a Long Pending state, a timer (t_long) starts. §VII.E.2; APPLE-1005, [0098], FIG. 3. “If the timeout expires, a long press event is generated and the state machine enters the Repeat Pending or Press Complete states,” as Bowens describes, which a POSITA would have understood as corresponding to Kim’s “tap and hold” or “long press,” which causes the KBD to *invoke a continuous zoom-in or zoom-out function*. APPLE-1004, [0104]-[0107], [0139]-[0140], FIG. 7; APPLE-1005, [0098]. While Figure 7 shows the object (person on bicycle) at reference numbers 741, 743, and 745 increasing in size (e.g., zoom in), Kim explains that “the hold event may be alternatively assigned to a *continuous zoom-out function*” such that the tap and hold input causes the object (person on bicycle) to decrease in size (e.g., zoom out). *Id.*, [0107] (emphasis added), FIG. 7. Modified Figure 7 (below) illustrates Kim’s alternative assignment.



APPLE-1004, FIG. 7 (“hold” scenario modified)

206. A POSITA would have understood that a “continuous zoom-out function” on an object, such as the bike and bicyclist of Kim’s Figure 7, responsive to a persistent gesture (i.e., a gesture with a constantly changing duration because it is ongoing), is the equivalent of scale operation that includes a reduction in size. Kim explains that, when the hold event is assigned to a zoom-in process, “the mobile device displays an enlarged display target data.” APPLE-1004, [0106]. “In such a zoom-in process, the mobile device continuously offers real-time, scaled-up

bitmap image on the screen in response to the duration of the hold event.” *Id.*, [0105]. A POSITA would have understood that, when the hold event is assigned to a zoom-out process, the mobile device would instead display a scaled-down bitmap image on the screen in response to the duration of the hold event. APPLE-1004, [0031] (“a gesture refers to a touch-based user’s input motion which travels in a given pattern assigned to a zoom-in or zoom-out function in a zoom mode”), [0106] (“the hold event may be alternatively assigned to a continuous zoom-out function”).

207. Additionally, Kim teaches that the zoom-out is “continuous”: in a zoom-out process, “the mobile device continuously offers a real-time, scaled-[down] bitmap image on the screen in response to the duration of the hold event.” APPLE-1004, [0105]. Thus, the longer the duration of the gesture (press and hold), the greater the zoom-out, or the more the size is reduced. *Id.* Accordingly, to the extent this claim is interpreted to require a reduction in size, wherein the size is inversely related to the duration, Kim discloses the limitation. *Id.*

208. In the alternative, if the phrase “where the scale operation includes a reduction in a size that is inversely related to the duration” is interpreted to mean that the *reduction* and the *duration* are inversely related (as opposed to the *size* and the *duration* being inversely related), then such an element would have at least been obvious. For example, Kim teaches that a double tap event (i.e., short

duration) causes “enlarged display target data” to be “given basic magnification value” (i.e., reduction in size), which a hold event (i.e., long duration) causes a zoom-out function (i.e., reduction in size). APPLE-1004, [0105]-[0107]. That is, in the case of a double tap, if the “enlarged display target data” was very large, the double tap would cause a large reduction in size, whereas, a long press that was relatively short would cause a smaller reduction in size. *Id.* Thus, the scale operation is responsive to a change in a duration of the gesture (e.g., from a short duration to a long one). *Id.* Thus, the gesture with the longer duration of the gesture (long press) would cause a comparatively smaller reduction in size. *Id.* In this way, the *reduction* in a size is inversely related to the *duration* of the gesture (the larger the duration of the gesture, the less the reduction). *Id.*

209. In another alternative, if the phrase “where the scale operation includes a reduction in a size that is inversely related to the duration” is interpreted to mean that either the *reduction in size* or the *rate of reduction in size* are inversely related *duration*, then such an element would have at least been obvious. In particular, a POSITA would have understood that as an image is reduced in size (as zoomed out over time), fewer pixels are changed per step in the zoom-out animation (e.g., a step change from 100% to 99% zoom would change a larger number of pixels than a step change from 2% to 1%). *Id.* In this way, the *reduction* in a size is inversely related to the *duration* of the gesture. *Id.*

210. Finally, while the example in Figure 7 shows an object (rider on a bicycle) with a surrounding interface that zooms in or zooms out in response to the press and hold, Figure 8 shows yet another example of the zoom-in or zoom-out effect. APPLE-1004, [0111]-[0115], FIG. 8. In particular, Figure 8 shows an example where, in response to a user gesture, both an object (e.g., a car) and its surrounding interface (e.g., the remainder of the web page surrounding the object), a zooming action is performed on both the object and its surrounding interface. *Id.* Thus, Kim's scale operation is performed in connection with the at least portion of the interface. *Id.*

5. Claim 32

[32.pre] An apparatus, comprising:

211. *See discussion of [32.pre] (Ground 1A).*

[32.a] at least one non-transitory memory storing instructions and a plurality of applications;

212. *See discussion of [32.a] (Ground 1A).*

[32.b] a touch screen; and

213. *See discussion of [32.b] (Ground 1A).*

[32.c] 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 the instructions to:

214. *See discussion of [32.c] (Ground 1A).*

[32.d] display, via the touch screen, a first object and at least one other object;

215. The KBD renders [32.d] obvious. As discussed above with respect to [27.d], the KBD displays a first object. APPLE-1004, FIG. 7. To the extent the device shown in Kim's Figure 7 is interpreted as a single object, a POSITA would have found the display of at least one other object obvious and understood Figure 7 to simply be an example. Figure 8 of Kim, for example, shows more than one object (e.g., a car, text, a person). APPLE-1004, FIG. 8. Thus, a POSITA would have found the display of any number of objects to be obvious.

[32.e] detect at least one component of a gesture on the touch screen on the first object;

216. The KBD renders [32.e] obvious. *See* discussion of [32.e] (Ground 1A). A POSITA would have understood or found obvious that the gesture detected on the KBD's touch screen includes a plurality of time-related components, including, for example, how long the gesture has been maintained at any point in time during or at cessation of the gesture. The significance of these time-related components is clear in both Kim's and Bowens's teachings, which each define various types of gestures based on the length of the gesture. *See* APPLE-1004, [0034] ("*long press*"), [0061] ("duration of the hold event"), [0105] ("*duration of the hold event*"), [0139] ("*long press*"); APPLE-1005, [0023] ("touch of a *duration longer* than a threshold *duration*") ("a very *long press*"), [0029] ("if a touch of a *duration longer* than a threshold *duration* is sensed") ("distinguish between

different *durations* of press”), [0056]-[0059] (“*short touch duration*”), [0080] (“*short duration* gestures such as flicks and taps, and *long duration* gestures such as a press”), [0096] (“multiple interpretations of a single touch depending on its *duration*”).

[32.f] when a first component of the gesture is detected where a magnitude of a duration of the gesture being detected on the touch screen on the first object is less than a first magnitude threshold, perform a function upon a cessation of the gesture being detected;

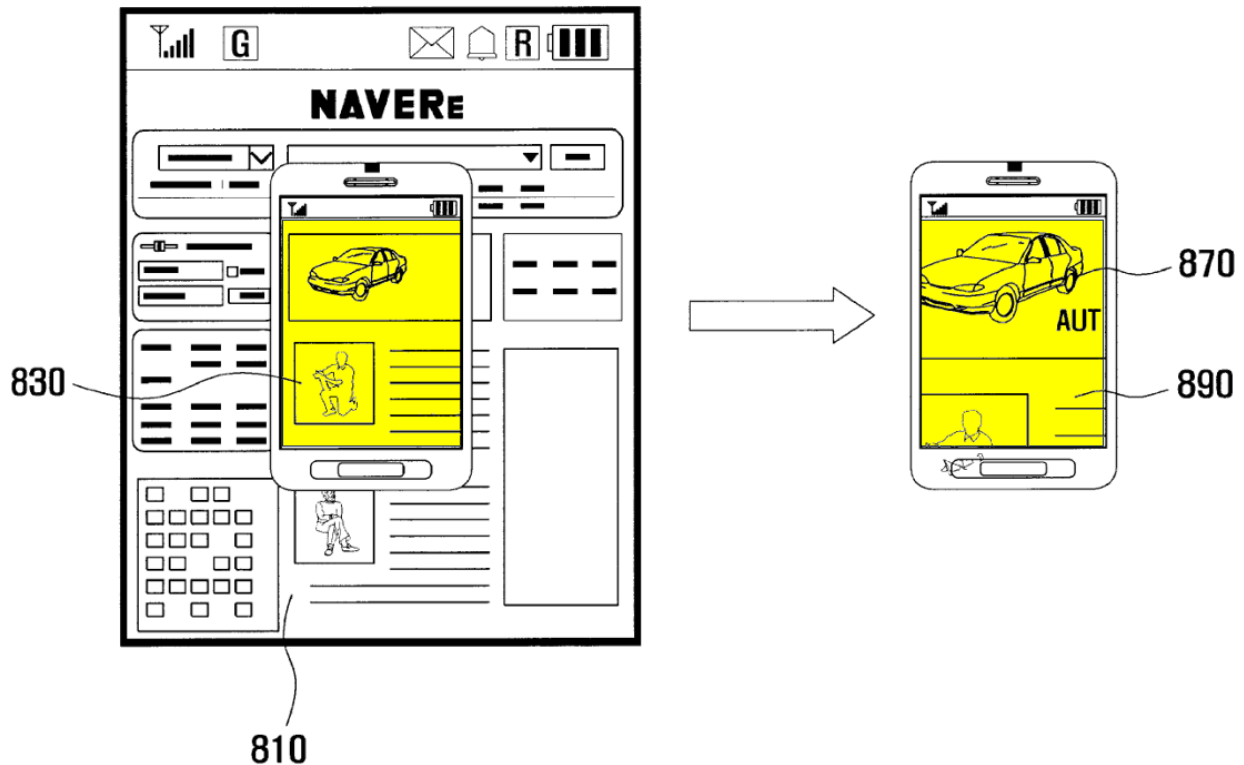
217. The KBD renders [32.f] obvious for at least the same reasons as discussed for [1.f] (Ground 2A). In particular, when a first component of the gesture (time-related component when a double tap event is generated) is detected where the duration detected on the touch screen is less than a first magnitude (e.g., the timeout for the transition from Touched to Pressed), the KBD performs a function (displays target data with a given basic magnification value). See discussion of [1.f] (Ground 2A). Moreover, a POSITA would have understood that, where the function occurs after a timeout, shortly after cessation of the gesture (when the user releases his finger), the function is performed upon cessation of the gesture being detected. APPLE-1004 (“Thereafter, a user can invoke a double tap event on the screen. Then, *in response to a double tap event*, the mobile device displays target data with a given basic magnification value...”).

[32.g] when a second component of the gesture is detected where the magnitude of the duration of the gesture being detected on the touch screen on the

first object is greater than the first magnitude threshold, perform a scale operation for the at least one other object,

218. The KBD renders [32.g] obvious for at least the same reasons as discussed for [1.g] (Ground 2A). In particular, when a second component of the gesture (time-related component when a long press event is generated) is detected where the duration detected on the touch screen is greater than the first magnitude (the timeout for the transition from Touched to Pressed), the KBD performs a function (a zoom or scale operation). *See* discussion of [1.g] (Ground 2A). Moreover, a POSITA would have understood that the scale operation is performed on content displayed on the screen, including both the first object (hyperlink) and the at least one other object (e.g., car, person). Figure 8 of Kim (below) shows an example showing how all of the displayed content (highlighted) is scaled in response to the gesture. APPLE-1004, FIG. 8.

FIG. 8



APPLE-1004, FIG. 8 (highlighted)

[32.h] where the scale operation includes a reduction in a size so as to be inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the first object; and

219. See discussion of [1.h] (Ground 2A).

[32.i] when a third component of the gesture is detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is greater than a second magnitude threshold, perform another function.

220. The KBD renders [32.i] obvious for at least the same reasons as discussed for [1.i] (Ground 2A). In particular, when a third component of the gesture (e.g., time-related component when a tap event is generated) is detected

where the duration detected on the touch screen is greater than a second magnitude (the “timeout/tap” transition from SecondTapPending to Idle), the KBD performs another function (activating the hyperlink). *See* discussion of [1.i] (Ground 2A).

F. Ground 2B: Claims 34, 59, and 64 are obvious over Kim in view of Bowens and Lang

1. The Kim-Bowens-Lang Combination

221. The Kim-Bowens Device is discussed above in §VII.E.2. For at least the same reasons discussed in §VII.B.2, it would have further been obvious to a POSITA to implement the KBD in a smart watch, as taught by Lang, forming the Kim-Bowens-Lang Device (“KBLD”).

2. Claim 34

[34.pre] A method, comprising:

222. To the extent the preamble is limiting, the KBLD renders [34.pre] obvious. In particular, as discussed with respect to claim 1, the KBD (apparatus) performs a method. §VII.E.2. For at least the same reasons, the KBLD performs a method, as discussed in more detail with respect to [34.e]-[34.j]. *See* discussion of [1.pre] (Ground 2A).

[34.a] at an apparatus including a watch with

223. The KBLD renders [34.a] obvious. Kim explains that its mobile device can be “any other kinds of portable and handheld devices.” APPLE-1004,

[0118]. As discussed above in §VII.F.1, a POSITA would have found it obvious to implement the KBLD as a smart watch, as taught by Lang. APPLE-1005, [0015].

[34.b] at least one non-transitory memory storing a plurality of applications,

224. See discussion of [1.a] (Ground 2A).

[34.c] a touch screen, and

225. See discussion of [1.b] (Ground 2A).

[34.d] one or more processors in communication with the at least one non-transitory memory and the touch screen:

226. See discussion of [1.c] (Ground 2A).

[34.e] displaying, via the touch screen, an object and at least a portion of an interface;

227. See discussion of [1.d] (Ground 2A).

[34.f] detecting a gesture on the touch screen on the object;

228. See discussion of [1.e] (Ground 2A).

[34.g] based on a magnitude of a duration of the gesture detected on the touch screen on the object being detected to be less than a first magnitude threshold, performing an operation;

229. See discussion of [1.f] (Ground 2A).

[34.h] based on the magnitude of the duration of the gesture detected on the touch screen on the object being detected to be greater than the first magnitude threshold, performing a scaling in connection with at least the object,

230. See discussion of [1.g] (Ground 2A).

[34.i] where the scaling includes a reduction in a size such that the size is inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the object; and

231. See discussion of [1.h] (Ground 2A).

[34.j] based on the magnitude of the duration of the gesture detected on the touch screen on the object being detected to be greater than a second magnitude threshold, performing another operation.

232. See discussion of [1.i] (Ground 2A).

3. Claim 59

[59.pre] A method, comprising:

233. See discussion of [34.pre] (Ground 2B).

[59.a] at an apparatus including a watch with

234. See discussion of [34.a] (Ground 2B).

[59.b] at least one non-transitory memory storing a plurality of applications,

235. See discussion of [34.b] (Ground 2B).

[59.c] touch screen, and

236. See discussion of [34.c] (Ground 2B).

[59.d] one or more processors in communication with the at least one non-transitory memory and the touch screen:

237. See discussion of [34.d] (Ground 2B).

[59.e] displaying an object on at least a portion of an interface;

238. See discussion of [27.d] (Ground 2A).

[59.f] detecting a gesture via the touch screen on the object;

239. See discussion of [27.e] (Ground 2A).

[59.g] performing a first function in a first touch state;

240. See discussion of [27.f] (Ground 2A).

[59.h] performing a second function in a second touch state; and

241. See discussion of [27.g] (Ground 2A).

[59.i] based on a detection of at least a portion of the gesture and based on a change in a duration of the gesture being detected via the touch screen on the object, performing a scale operation in connection with the at least portion of the interface, where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected via the touch screen on the object

242. See discussion of [27.h] (Ground 2A).

4. Claim 64

[64.pre] A method, comprising:

243. See discussion of [34.pre] (Ground 2B).

[64.a] at an apparatus including a watch having

244. See discussion of [34.a] (Ground 2B).

[64.b] at least one non-transitory memory storing a plurality of applications,

245. See discussion of [34.b] (Ground 2B).

[64.c] a touch screen, and

246. See discussion of [34.c] (Ground 2B).

[64.d] one or more processors in communication with the at least one non-transitory memory and the touch screen:

247. See discussion of [34.d] (Ground 2B).

[64.e] displaying, via the touch screen, a first object and at least one other object;

248. See discussion of [32.d] (Ground 2A).

[64.f] detecting at least one component of a gesture on the touch screen on the first object;

249. See discussion of [32.e] (Ground 2A).

[64.g] in response to a first component of the gesture being detected where a magnitude of a duration of the gesture being detected on the touch screen on the first object is detected to be less than a first magnitude threshold, performing a function upon a cessation of the gesture being detected;

250. See discussion of [32.f] (Ground 2A).

[64.h] in response to a second component of the gesture being detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is detected to be greater than the first magnitude threshold, performing a scale operation for the at least one other object,

251. See discussion of [32.g] (Ground 2A).

[64.i] where the scale operation includes a reduction in a size so as to be inversely related to the magnitude of the duration of the gesture being detected on the touch screen on the first object; and

252. See discussion of [32.h] (Ground 2A).

[64.j] in response to a third component of the gesture being detected where the magnitude of the duration of the gesture being detected on the touch screen on the first object is detected to be greater than a second magnitude threshold, performing another function.

253. See discussion of [32.i] (Ground 2A).

**G. Ground 2C: Claim 30 is obvious over Kim in view of
Bowens and Burrough**

1. The Kim-Bowens-Burrough Combination

254. Kim describes a mobile device that “has a touch sensitive surface such as a touch screen” that provides “a touch-based zooming control method.”

APPLE-1004, Abstract. Like many devices with touch sensitive surfaces, however, Kim does not teach that any tactile feedback is provided to a user, despite haptic feedback being well known by the Critical Date. For instance, Lacroix explains:

Electronic device manufacturers strive to produce a rich interface for users. Conventional devices use visual and auditory cues to provide feedback to a user. In some interface devices, kinesthetic feedback (such as active and resistive force feedback) and/or tactile feedback (such as vibration, texture, and heat) is also provided to the user, more generally known collectively as “haptic feedback.” Haptic feedback can provide cues that enhance and simplify the user interface. Specifically, vibration effects, or vibrotactile haptic effects, may be useful in providing cues to users of electronic devices to alert the user to specific events, or provide realistic feedback to create greater sensory immersion within a simulated or virtual environment.

APPLE-1008, [0003]. The lack of a physical keyboard or tactile means for providing input can prove challenging to a user, especially in scenarios of poor lighting, small screens, or large fingers. As such, a POSITA would have been motivated to look to references, such as Burrough, which teaches the use of haptic feedback in conjunction with touch screens. APPLE-1007, Abstract.

255. Indeed, Burrough recognizes that “physical sensation can not be provided by a conventional multi-touch system thereby forcing the user to visually

locate the home key thereby making keyboard use less efficient and fatiguing.”

APPLE-1007, [0008]. “[R]esearch has shown that providing the multi-touch surface with the ability to provide physical (haptic) feedback makes the multi-touch experience even more efficient and realistic to the user.” *Id.* Thus,

Burrough teaches a portable device that provides haptic feedback in response to user input, including zooming operations. *Id.*, [0079]-[0081], FIG. 11. The haptic feedback is provided by way of a haptic device (e.g., piezoelectric actuator assembly) in the portable device that provides vibro-tactile response to the user.

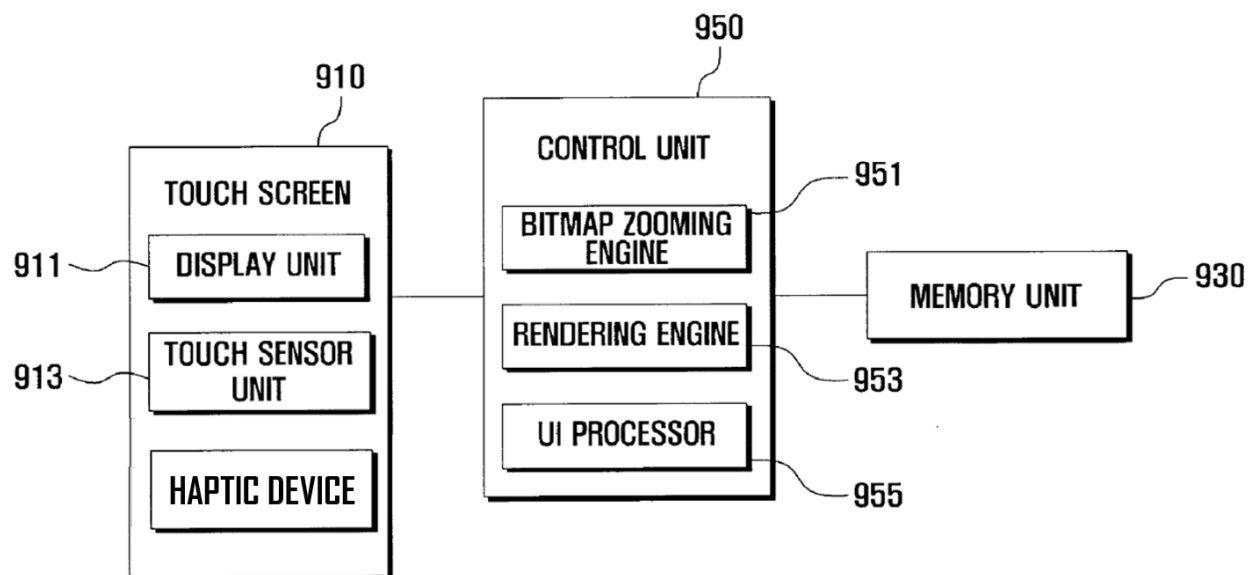
Id., [0013], [0033], [0037], [0047], [0055]-[0056], FIGS. 1A-1E

256. A POSITA would therefore have been motivated to include a haptic device and use of haptic feedback, as Burrough teaches and suggests, in the KBD to provide haptic feedback to improve the user experience with the touch screen.

In particular, a POSITA would have been motivated to provide haptic feedback to a user invoking a zoom operation, as Burrough teaches, for example, to provide an indication as to whether a zoom-in or a zoom-out gesture has been invoked.

APPLE-1007, [0080]. As shown in Figure 1B of Burrough, the haptic actuator 136 is in communication with a micro-controller 132 and processor 106 of the portable device. APPLE-1007, [0046] (“Although micro-controller 132 is shown as a component separate from processor 106, it is contemplated that functions carried out by micro-controller 132 can in fact be performed by processor 106.”), [0047]

(“cause microcontroller 132 to active at least haptic actuator 136”), FIG. 1B; In the Kim-Bowens-Burrough Device (“KBBD”), the haptic device would be in communication with the processor (“control unit 950”) of Kim, such that the control unit 950 could control haptic device and its generating of haptic responses on the touch screen, as illustrated below in modified Figure 9 from Kim.



APPLE-1004, FIG. 9 (modified)

257. Combining Kim and Bowens (*see* §VII.E.2) with Burrough would have been obvious at least because such a combination would have merely involved (1) combining prior art elements according to known methods to yield predictable results and (2) the use of known technique to improve similar devices (methods, or products) in the same way.

2. Claim 30

[30.pre] An apparatus, comprising:

258. See discussion of [1.pre] (Ground 1A).

[30.a] at least one non-transitory memory storing instructions and a plurality of applications;

259. See discussion of [1.a] (Ground 1A).

[30.b] a touch screen;

260. See discussion of [1.b] (Ground 1A).

[30.c] a vibrator; and

261. The KBBD renders [30.c] obvious. As discussed above in §VII.G.1 and §VII.C.2, the KBBD further includes a haptic device (e.g., piezoelectric actuator assembly), which provides a “vibro-tactile response,” and thus is a vibrator. APPLE-1007, [0013], [0033], [0037], [0047], [0055]-[0056], FIGS. 1A-1E; §§VII.C.2, VII.G.1.

[30.d] one or more processors in communication with the at least one non-transitory memory, the touch screen, and the vibrator, wherein the one or more processors execute the instructions to cause the apparatus to:

262. The KBBD renders [30.d] obvious with respect to the one or more processors in communication with memory and the touch screen for at least the same reasons as discussed for [1.c]. As discussed above in §VII.G.1 and §VII.C.2, the KBBD further includes a vibrator in communication with the control unit of Kim. §§VII.C.2, VII.G.1.

263. As discussed above in §VII.C.2 and §VII.G.1, the KBBD further includes a vibrator in communication with the one or more processors. §VII.C.2, §VII.G.1.

[30.e] display a first object and at least one other object;

264. See discussion of [32.d] (Ground 2A).

[30.f] detect a gesture on the touch screen on the first object;

265. See discussion of [1.e] (Ground 2A).

[30.g] during detection of at least a portion of the gesture before a completion thereof is detected, perform a scale operation, based on a change in a duration of the gesture being detected on the touch screen on the first object, on at least the at least one other object,

266. See discussion of [30.g] (Ground 1C).

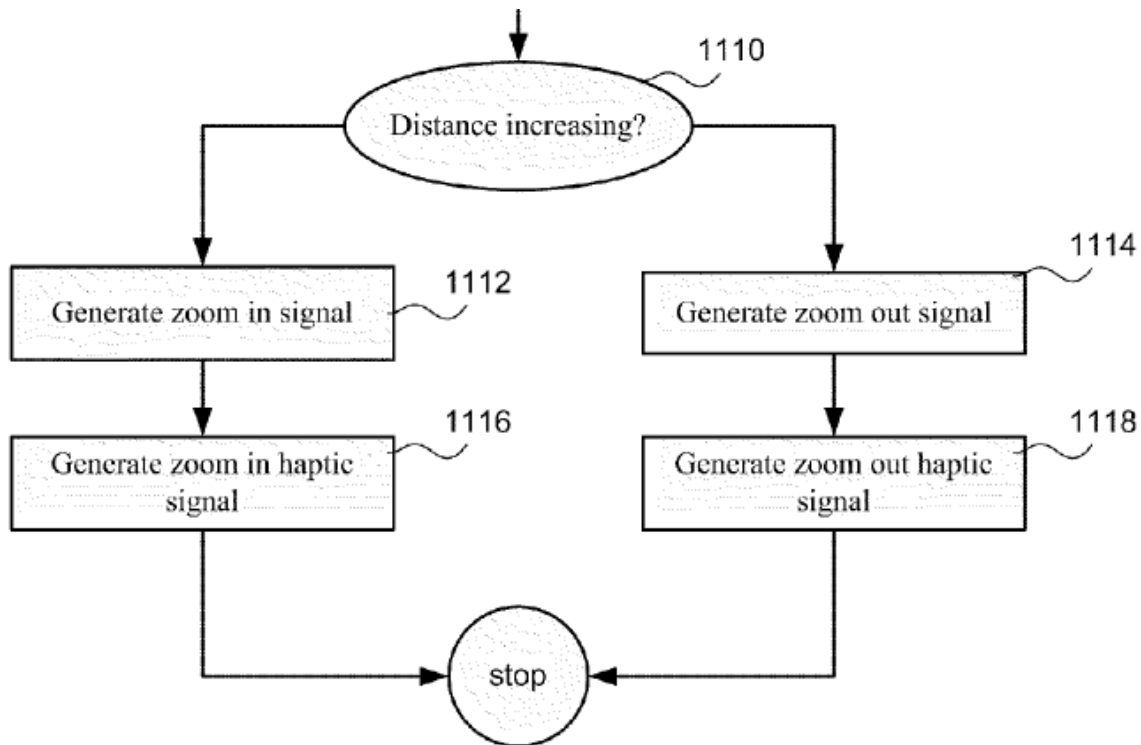
[30.h] where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected on the touch screen on the first object; and

267. See discussion of [32.h] (Ground 2A).

[30.i] after at least a portion of the scale operation, vibrate the apparatus and perform an action.

268. The KBBD renders [30.i] obvious. As discussed above in §§VII.C.2 and VII.G.1, the KBBD includes a haptic device (e.g., piezoelectric actuator assembly) that provides a “vibro-tactile response.” APPLE-1007, [0013], [0033], [0037], [0047], [0055]-[0056], FIGS. 1A-1E, FIG. 11; §§VII.C.2, VII.G.1. As shown in Figure 11 (excerpted below), Burrough teaches “generate a zoom out

haptic signal” or ”generate a zoom in haptic signal” after generating a zoom out or zoom in signal. APPLE-1004, [0080], FIG. 11.



APPLE-1007, FIG. 11 (excerpted)

269. Additionally, as explained in §VII.E.2, in the KBD (which Burrough modifies), as the gesture persists and the timeout expires, “the state machine enters the Repeat Pending or Press Complete states” (perform an action), as Bowens describes. APPLE-1005, [0100].

H. Ground 2D: Claim 62 is obvious over Kim in view of Bowens, Lang, and Burrough

1. The Kim-Bowens-Lang-Burrough Combination

270. The Kim-Bowens-Lang Device (KBLD) is discussed above in §VII.F.1. For at least the same reasons discussed in §VII.G.1, it would have

further been obvious to a POSITA to modify the KBLD to include a haptic device and use of haptic feedback, as Burrough teaches and suggests.

1. Claim 62

[62.pre] A method, comprising:

271. See discussion of [34.pre] (Ground 2B).

[62.a] at an apparatus including a watch

272. See discussion of [34.a] (Ground 2B).

[62.b] having at least one non-transitory memory storing a plurality of applications,

273. See discussion of [34.b] (Ground 2B).

[62.c] a touch screen,

274. See discussion of [30.b] (Ground 2C).

[62.d] a vibrator, and

275. See discussion of [30.c] (Ground 2C).

[62.e] one or more processors in communication with the at least one non-transitory memory, the touch screen, and the vibrator:

276. See discussion of [30.d] (Ground 2C).

[62.f] displaying a first object and at least one other object;

277. See discussion of [30.e] (Ground 2C).

[62.g] detecting a gesture on the touch screen on the first object;

278. See discussion of [30.f] (Ground 2C).

[62.h] in response to a detection of at least a portion of the gesture, performing a scale operation, based on a change in a duration of the gesture being

detected on the touch screen on the first object, on at least the at least one other object,

279. See discussion of [30.g] (Ground 2C).

[62.i] where the scale operation includes a reduction in a size that is inversely related to the duration of the gesture being detected on the touch screen on the first object; and

280. See discussion of [30.h] (Ground 2C).

[62.j] subsequent to the performance of the scale operation, vibrating the apparatus and performing an action.

281. See discussion of [30.i] (Ground 2C).

APPENDIX A

Andy Cockburn

B.Sc.(Hons) Ph.D.

Curriculum Vitae

Full legal name	Andrew Jeremy Gavin Cockburn
University Address	Department of Computer Science and Software Engineering University of Canterbury Christchurch 8140 New Zealand
Telephone	+64 3 369 2119
Fax	+64 3 364 2569
Email	andy.cockburn@canterbury.ac.nz
URL	http://www.csse.canterbury.ac.nz/~andy
Consultancy Address	Cockburn Human Interface Consulting Ltd. 20 Gilmour Terrace, Lyttelton 8082 New Zealand
Telephone	+64 21 296 2307
Email	ajc168@snap.net.nz
Date of birth	23rd July 1966
Date	June 24, 2024

Brief Summary

International research reputation

- Over one hundred and ninety publications in international journals and conferences.
- Elected to the CHI Academy (2015), which is an honorary group of individuals who have made substantial contributions to the field of human-computer interaction.
- Recipient of the 2009 Chris Wallace award for the outstanding research contribution to Computer Science (2005-2007) in Australasia.
- Deputy Chair of the national panel for assessing Computer Science and Mathematics research quality (MIST PBRF panel, 2016-2019).
- Technical programme committee chair for ACM CHI 2020; Papers co-chair for ACM CHI 2014 and 2015.
- Editorial Board Member for: ACM Transactions on Computer-Human Interaction (2012-2021); Human-Computer Interaction Journal (2015-); Foundations and Trends in HCI (2015-2023); International Journal of Human-Computer Studies (2004-2013); Interacting with Computers (2003-2013).
- Best paper awards at CHI 2018, 2013, 2012, 2011 (awarded to the top 1% of submissions) and 13 Honourable Mention awards (top 5%) at ACM CHI.
- H-index impact factor of 56 (Google Scholar).

Excellent educator and public speaker

- Long track record of outstanding student evaluations.
- Nominated for the New Zealand National Tertiary Teaching Award, 2010. Recipient of a University of Canterbury Teaching Award (2004) and many nominations for student-elected 'lecturer of the year' award.

Experienced expert witness and consultant

- Testifying expert witness on behalf of industry leaders in various jurisdictions, including the Federal Court of Australia, US Federal Court, US Patent Trial and Appeal Board, and the US International Trade Commission.

Education

Degree	PhD (June 1993)
Discipline	Computer Science — Computer Supported Cooperative Work
Location	University of Stirling, Scotland
Thesis Title	Groupware Design: Principles, Prototypes, and Systems

Degree	BSc(Hons) (July 1988)
Discipline	Computer Science
Location	University of York, England

Employment History

Position	Professor (promoted through lecturer, senior lecturer, associate professor)
Location	University of Canterbury Christchurch New Zealand
Period	June 1993 — current

Research Student Supervision

PhD candidates

Stewart Dowding	2021–	Interaction Pace and User Preferences
Pang Suwanaposee	2020–	Cognitive Biases and the Subjective Experience of Interaction
Sarmad Soomro	2017–2021	Interaction with In-Vehicle Touchscreens
Joshua Leung	2013–2017	Understanding the Noticeability and Distraction of Interactive Highlighting Techniques
Philip Quinn	2012–2016	Economic Behaviour & Psychological Biases in Human-Computer Interaction
Tham Piumsomboon	2011–2015	Adaptive and Multimodal Interfaces for Collaborative Augmented Reality
Joey Scarr	2011–2014	Improving Interaction with Spatial User Interfaces
Stephen Fitchett	2010–2013	Understanding and Improving Personal File Retrieval
Susanne Tak	2008–2011	Improving Interfaces for Window/Task Switching
Jason Alexander	2006–2009	Improving Interfaces for Document Navigation
Joerg Hauber	2004–2008	Understanding Remote Collaboration in Video Collaborative Virtual Environments
Julian Looser	2003–2007	AR Magic Lenses: Addressing the Challenge of Focus and Context in Augmented Reality
Volkert Buchmann (A. sup.)	2004–2008	Haptic Feedback in Augmented Reality
Carl Cook (A. sup.)	2002–2006	Collaborative Software Engineering
Tim Wright	2000–2004	Collaborative and Multiple-Notation Programming Environments for Children
Michael JasonSmith	2000–2006	User Interfaces for Temporal Awareness

M.Sc. candidates

Ben McDonald	2010–2011	Multuser interaction with large public displays
Dominic Winkler	2009–2011	Interaction Techniques for Cross Context Activities
Taher Amer	2005–2006	Evaluating SwiftPoint
Trond Nilsen	2004–2006	Hybrid Interfaces in Augmented Reality
Joshua Savage	2003–2004	The Calibration and Evaluation of Speed-Dependent Automatic Zooming Interfaces
Ben Schmidt	2001–2002	A Study of Bimanual Interaction
Elizabeth Ng (A. sup.)	1997–1998	Pen-based music input
Aaron Tay	1994–5	End-User Tailorability in Workflow Design

Honours candidates and Honours-level interns

Joel Harrison	2012	Improving Command Selection
Joshua Leung	2012	Improving File Access
Jerome Delamarche	2010	Multiple Trajectory Pointing Methods
Joey Scarr	2010	Improving Expertise Development in User Interfaces
Stephen Fitchett	2009	Improved Mobile Scrolling Interfaces
Philip Quinn	2007	Improving Command Interfaces
Keith Humm	2007	Improving Interfaces for Task Switching
Jason Alexander	2005	Improving Document Navigation with Space-Filling Thumbnails
Philip Brock	2005	An Investigation of Target Acquisition with Visually Expanding Targets in Constant Motor-Space
David Mitchell	2003	Evaluating Focus+Context Screens
Andrew Wallace	2003	Calibrating Speed-Dependent Automatic Zooming
Andrew Barrett	2002	Implementing RSVP as an Image Browser
Julian Looser	2002	3D Games as Motivation in Fitts' Law Experiments
Joshua Savage	2002	Implementing and Evaluating Speed-Coupled Scrolling
Amal Siresena	2002	Mobile Text Entry
Michael Moyle	2001	An Evaluation of Simple Gesture Based Controls
Lee Butts	2001	Mobile Phone Text Entry
Eddie Edwards	2001	Towards Scalable Interfaces Using Spatial Cues
Gene Thomas	1997	Smart Desktop
Olof Eilert	1997	User Interfaces for Object Oriented Literate Programming
Philip Weir	1996	Distortion Oriented Workspace Awareness
Carey Evans	1996	Assisting World Wide Web Navigation
Justin Macfarlane	1995	Mawd: A Collaborative Analysis Tool
Carl Cerecke	1995	TRAIL: An Email Based CSCW Toolkit
Andrew Bryant	1995	Leogo: An Equal Opportunity Programming Environment
Ian Bosely	1994	SPI: A Social Presence Indicator

Postgraduate Thesis Examinations

Katherine Fennedy	PhD	External Examiner	Singapore University of Tech. & Design	2021
Praveenkumar Kanithi	PhD	Chair	University of Canterbury	2020
Ori Gagoni	PhD	Chair	University of Canterbury	2019
Alaeddin Nassani	PhD	Chair	University of Canterbury	2019
Veera Mandalika	PhD	Chair	University of Canterbury	2018
Jessalyn Alvina	PhD	Examiner	Université Paris Sud	2017
Hyungon Kim	PhD	Chair	University of Canterbury	2017
Thomas Young	PhD	Chair	University of Canterbury	2017
Craig Sutherland	PhD	Examiner	University of Auckland	2016
David Thompson	PhD	Chair	University of Canterbury	2016
Philip Quinn	PhD	Supervisor	University of Canterbury	2016
Daniel Bradley	PhD	Examiner	University of Queensland	2016
David Altimera	PhD	Chair	University of Canterbury	2015
Jakub Zlotowsky	PhD	Chair	University of Canterbury	2015
Thammathip Piumsomboon	PhD	Supervisor	University of Canterbury	2015
Robert Schattschnieder	PhD	Chair	University of Canterbury	2014
Hazel Bradshaw	PhD	Chair	University of Canterbury	2014
Amur Al Manji	PhD	Examiner	University of Auckland	2014
Amir Shareghi Najar	PhD	Chair	University of Canterbury	2014
Joey Scarr	PhD	Supervisor	University of Canterbury	2014
Stephen Fitchett	PhD	Supervisor	University of Canterbury	2014
Amali Weeseringhe	PhD	Chair	University of Canterbury	2012
Sayan Ray	PhD	Chair	University of Canterbury	2012
Christina Dicke	PhD	Chair	University of Canterbury	2012
Minkyung Lee	PhD	Chair	University of Canterbury	2010
Jason Alexander	PhD	Supervisor	University of Canterbury	2009
Leah Findlater	PhD	Examiner	University of British Columbia	2009
Judy Bowen	PhD	Examiner	University of Waikato	2008
Joerg Hauber	PhD	Supervisor	University of Canterbury	2008
Julian Looser	PhD	Supervisor	University of Canterbury	2008
Michael JasonSmith	PhD	Supervisor	University of Canterbury	2006
Tim Wright	PhD	Supervisor	University of Canterbury	2005
Beryl Plimmer	PhD	Examiner	University of Waikato	2003
Gorden Paynter	PhD	Examiner	University of Waikato	2000
Gitesh K Raikundalia	PhD	Examiner	Bond University	1997
Ramon Craig Standing	PhD	Examiner	University of Western Australia	1997
Ying Leung	PhD	Examiner	Massey University	1995
Qi Min Ser	Masters	Examiner	University of Canterbury	2016
Taher Amer	Masters	Supervisor	University of Canterbury	2006
Trond Nilsen	Masters	Supervisor	University of Canterbury	2006
Akhil Mehra	Masters	Examiner	University of Auckland	2005
Aparna Krishnan	Masters	Examiner	University of Waikato	2003
Shaun Kaasten	Masters	Examiner	University of Calgary	2001
Boyd Ludlow	Masters	Examiner	University of Waikato	1999
Donald Cox	Masters	Examiner	University of Calgary	1998
Wallace Chigona	Masters	Examiner	Waikato University	1997
Simon Gianoutsos	Masters	Examiner	Waikato University	1997
Jing Deng	Masters	Examiner	Waikato University	1996 and 1997
Aaron Tay	Masters	Examiner	University of Canterbury	1996

Student glory

Pang Suwanaposee	PhD	2022	Google PhD Fellowship Award AUD\$15,000.
Pang Suwanaposee	PhD	2021	CHI Honourable Mention award.
Joey Scarr	PhD	2016	John Makepeace Bennett Award for the Distinguished Computer Science PhD thesis in Australasia.
Philip Quinn	PhD	2015	US Patent Awarded (Google Assignee).
Philip Quinn	PhD	2014	US Patent Awarded (Google Assignee).
Stephen Fitchett	PhD	2014	CHI Honourable Mention award.
Philip Quinn	PhD	2013	Google Mountain View Internship.
Stephen Fitchett	PhD	2013	CHI Best Paper award.
Joey Scarr	PhD	2013	CHI Honourable Mention award.
Joey Scarr	PhD	2013	CHI ReplicCHI award.
Joey Scarr	PhD	2012	CHI Best Paper award.
Stephen Fitchett	PhD	2012	CHI Honourable Mention award.
Philip Quinn	PhD	2012	Google Mountainview Internship.
Joey Scarr	PhD	2012	Google Sydney Internship.
Stephen Fitchett	PhD	2011	Outstanding reference from Darren Edge, Microsoft Research Asia.
Joey Scarr	PhD	2012	\$3K for 2nd place in University 'PhD in 3' competition.
Joey Scarr	PhD	2012	\$1K for 1st place in College 'PhD in 3' competition.
Stephen Fitchett	PhD	2012	\$1K for 3rd place in College 'PhD in 3' competition.
Philip Quinn	PhD	2011	Honourable Mention award ACM CHI'11.
Joey Scarr	PhD	2011	Google Sydney Internship.
Susanne Tak	PhD	2010	\$5K Google Anita Borg Scholarship
Susanne Tak	PhD	2010	\$3K from ACM for participation at CHI's Doctoral Consortium
Jason Alexander	PhD	2010	PhD Thesis placed on "Dean's list for excellence" (top 10%)
Susanne Tak	PhD	2009	\$2K award for 1st place in College Research Poster competition
Jason Alexander	PhD	2008	\$1K award for 2nd place in College Research Poster competition
Jason Alexander	PhD	2008	Michael Sweeney award for best student paper, Graphics Interface '08.
Mark Hancock	PhD	2007	Best paper nomination at ACM CHI'07.
Josh Savage and Andrew Wallace	M.Sc. Hons	2005	Best paper nomination at ACM CHI'05.
Mike Moyle	Hons	2005	NZ Hi-Tech Young Achiever of the Year Award

Publications

Edited Volumes

- EV1 J McGrenere and A Cockburn (Eds.) Proceedings of ACM CHI 2020 Conference on Human Factors in Computing Systems.
- EV2 J McGrenere and A Cockburn (Eds.) Extended Abstracts of ACM CHI 2020 Conference on Human Factors in Computing Systems.
- EV3 M Billinghamurst and A Cockburn (Eds.) Proceedings of the Sixth Australasian User Interface Conference (AUI2005). Volume 40 of the Conferences in Research and Practice in Information Technology, Australian Computer Society.
- EV4 A Cockburn (Ed.) Proceedings of the Fifth Australasian User Interface Conference (AUI2004). Volume 28 of the Conferences in Research and Practice in Information Technology, Australian Computer Society.

Papers in Refereed Journals and Collections

- J1 A Cockburn, D Hills, Z Che, and C Gutwin. User Interface Evaluation Through Implicit-Association Tests. *Proceedings of the ACM on Human-Computer Interaction*. EICS, Article v8eics262. 2024.
- J2 A Cockburn, A Goguey, C Gutwin, Z Chen, P Suwanaposee, and S Dowding. Automatically Adapting System Pace Towards User Pace – Empirical Studies. *International Journal of Human-Computer Studies*. 185:103228. May. 2024.
- J3 S Dowding, C Gutwin, and A Cockburn. User Speech Rates and Preferences for System Speech Rates. *International Journal of Human-Computer Studies*. 184:103222. April. 2024.
- J4 P Suwanaposee, C Gutwin, and A Cockburn. The Influence of Audio Effects and Attention on the Perceived Duration of Interaction. *International Journal of Human-Computer Studies*. 159:102756. March. 2022.
- J5 J Leung and A Cockburn. Design Framework for Interactive Highlighting Techniques. *Foundations and Trends in Human-Computer Interaction*. 14(2-3):96-271. 2021.
- J6 AF Mairena, C Gutwin, and A Cockburn. Which Emphasis Technique to Use? Perception of Emphasis Techniques with Varying Distractors, Backgrounds, and Visualization Types. *Information Visualization*. 21(2):95-129. 2022.
- J7 A Cockburn, P Dragicevic, L Besançon and C Gutwin. Threats of a Replication Crisis in Empirical Computer Science. *Communications of the ACM*. 63(8):70-79. August 2020.
- J8 M Claypool, A Cockburn and C Gutwin. The Impact of Motion and Delay on Selecting Game Targets with a Mouse. *ACM Transactions on Multimedia Computing, Communications, and Applications*. 16(2s):73, 1-14. June 2020.
- J9 P Quinn and A Cockburn. Loss Aversion and Preference in Interaction. *Human-Computer Interaction Journal*. 35(2):143-190. 2020. (Online from 2018).

- J10 A Cockburn, D Masson, C Gutwin, P Palanque, A Goguet, M Yung, C Gris and C Trask. Design and Evaluation of Braced Touch for Touchscreen Input Stabilisation. *International Journal of Human-Computer Studies*. 122:21-37. 2018.
- J11 B Lafreniere, C Gutwin, and A Cockburn. Investigating the Post-Training Persistence of Expert Interaction Techniques. *ACM Transactions on Human-Computer Interaction*. 24(4): Article 29 (46 pages). 2017.
- J12 A Cockburn, P Quinn and C Gutwin. The Effects of Interaction Sequencing on User Experience and Preference. *International Journal of Human-Computer Studies*. 108:89-104. 2017.
- J13 J Aceituno, S Malacria, P Quinn, N Roussel, A Cockburn and G Casiez. The Design, Use, and Performance of Edge-Scrolling Techniques. *International Journal of Human-Computer Studies*. 97:58-76. 2017.
- J14 J Scarr, C Gutwin, A Cockburn and A Bunt. StencilMaps and EphemeralMaps: Spatially Stable Interfaces that Highlight Command Subsets. *Behaviour and Information Technology*. 34(11):1092-1106. 2015.
- J15 S Fitchett and A Cockburn. An Empirical Characterisation of File Retrieval. *International Journal of Human-Computer Studies*. 74:1-13. 2015.
- J16 A Cockburn, C Gutwin, J Scarr and S Malacria. Supporting Novice to Expert Transitions in User Interfaces. *ACM Computing Surveys*. 47(2):31:1-36. 2014.
- J17 J Scarr, A Cockburn and C Gutwin. Supporting and Exploiting Spatial Memory in User Interfaces. *Foundations and Trends in Human-Computer Interaction*. 6(1):1-84. 2013.
- J18 P Quinn, A Cockburn and J Delamarche. Examining the Costs of Multiple Trajectory Pointing Techniques. *International Journal of Human-Computer Studies*. 71(4):492-509. 2013.
- J19 S Tak and A Cockburn. Enhanced Spatial Stability with Hilbert and Moore Treemaps. *Transactions on Visualization and Computer Graphics*. 19(1):141-148. 2013.
- J20 A Cockburn, D Ahlstrom and C Gutwin. Understanding Performance in Touch Selections: Tap, Drag, and Radial Pointing Drag with Finger, Stylus and Mouse. *International Journal of Human-Computer Studies*. 70(3):218-233. 2012.
- J21 J Hauber, A Cockburn, M Billinghamurst and H Regenbrecht. The Impact of Collaborative Style on the Perception of 2D and 3D Videoconferencing Interfaces. *Open Software Engineering Journal*. 6:1-20. 2012.
- J22 A Cockburn, P Quinn, C Gutwin, G Ramos and J Looser. Air Pointing: Design and Evaluation of Spatial Target Acquisition with and without Visual Feedback. *International Journal of Human-Computer Studies*. 69:401-414. 2011.
- J23 E Tanvir, A Bunt, A Cockburn and P Irani. Improving Cascading Menu Selections with Adaptive Activation Areas. *International Journal of Human-Computer Studies*. 69:769-785. 2011.
- J24 A Cockburn and C Gutwin. A Model of Novice and Expert Navigation Performance in Constrained Input Interfaces. *ACM Transactions on Computer-Human Interaction*. ACM Press. 7(3):1-38. 2010.

- J25 A Cockburn and C Gutwin. A Predictive Model of Human Performance with Scrolling and Hierarchical Lists. *Human-Computer Interaction Journal*. Taylor and Francis. 24(3): 273-314. 2009.
- J26 A Cockburn, A Karlson and B Bederson. A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. *ACM Computing Surveys*. ACM Press. 41(1): 1-31. 2008.
- J27 J Alexander, A Cockburn and R Lobb. AppMonitor: A Tool for Recording User Actions in Unmodified Windows Applications. *Behavior Research Methods*. Psychonomic Society. 40(2): 413-421. 2008.
- J28 G Casiez, D Vogel, R Balakrishnan and A Cockburn. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human-Computer Interaction Journal*. Taylor and Francis. 23(3): 215-250. 2008.
- J29 S Jones, M Jones, G Marsden, D Patel and A Cockburn. An Evaluation Of Integrated Zooming and Scrolling On Small Screens *International Journal of Human-Computer Studies*. 63(3): 271–303. Elsevier Science. 2005.
- J30 A Cockburn and S Brewster. Multimodal Feedback for the Acquisition of Small Targets *Ergonomics*. Taylor & Francis. 48(9): 1129–1150. 2005.
- J31 M Moyle and A Cockburn. A Flick in the Right Direction: A Case Study of Gestural Input *Behaviour and Information Technology*. 24(4): 275–288. Taylor & Francis. 2005.
- J32 A Cockburn. Web Browsers *Berkshire Encyclopedia of Human-Computer Interaction*. Bainbridge, WS (Editor). Volume 1: 80–83. Berkshire Publishing. 2004.
- J33 A Cockburn and B McKenzie. Evaluating Spatial Memory in Two and Three Dimensions *International Journal of Human-Computer Studies*. 61(3): 359–373. Elsevier Science. 2004.
- J34 A Cockburn and M Smith. Hidden Messages: Evaluating the Efficiency of Code Elision in Program Navigation *Interacting with Computers: The Interdisciplinary Journal of Human-Computer Interaction*. 15(3): 387–407. 2003.
- J35 A Cockburn, S Greenberg, S Jones, B McKenzie and M Moyle. Improving Web Page Revisitation: Analysis, Design, and Evaluation *Information Technology and Society: Special Issue on Web Navigation*, editors B Shneiderman, J Lazar, M Ivory. 3(1): 159–183. 2003.
- J36 A Cockburn, B McKenzie and M JasonSmith. Pushing Back: Evaluating a New Behaviour for the Back and Forward Buttons in Web Browsers *International Journal of Human-Computer Studies*. 57(5):397–414. Elsevier Science. 2002.
- J37 T Bell, A Cockburn, B McKenzie and J Vargo. Digital Lectures: If You Make Them, Will Students Use Them? Constraints on Effective Delivery of Flexible Learning Systems *International Multimedia Electronic Journal of Computer-Enhanced Learning*. 3(2). Association for the Advancement of Computing in Education. 2001 This paper was one of 25 (from 115 full papers) selected from the proceedings of ED-MEDIA 2001 for adaptation and inclusion in the journal.
- J38 A Cockburn. Supporting Tailorable Program Visualisations with Literate Programming and Fish-eye Views. *Information and Software Technology*. 43(13):745–758. Elsevier Science. 2001.

- J39 A Cockburn, T Bell and B McKenzie. Death to the Office of the Nineties: An HCI Perspective on Some Problems in Modern Office Information Systems *Journal of Research and Practice in Information Technology, Special Issue on Human-Computer Interaction (HCI)*., 33(1):68–82. Australian Computer Society. 2001.
- J40 A Cockburn and B McKenzie. What Do Web Users Do? An Empirical Analysis of Web Use. *International Journal of Human-Computer Studies*, 54(6):903–922. Academic Press. 2001.
- J41 A Cockburn and S Greenberg. Issues of Page Representation and Organisation in Web Browser’s Revisitation Tools. *Australian Journal of Information Systems*, 7(2):120–127. 2000.
- J42 A Cockburn and P Weir. An Investigation of Groupware Support for Collaborative Awareness Through Distortion-Oriented Views. *International Journal of Human Computer Interaction*, 11(3):231–255. Lawrence Erlbaum Associates. 1999.
- J43 A Cockburn and S Greenberg. The Design and Evolution of TurboTurtle, a Collaborative Microworld for Exploring Newtonian Physics. *International Journal of Human-Computer Studies*, 48(6):777–801. Academic Press. 1998.
- J44 A Cockburn and A Bryant. Leogo: An Equal Opportunity User Interface for Programming. *Journal of Visual Languages and Computing*, 8(5–6):601–619. Academic Press. 1997.
- J45 A Cockburn and S Jones. Which way now? Analysing and Easing Inadequacies in WWW Navigation. *International Journal of Human-Computer Studies*, 45(1):105–129. Academic Press. 1996.
- J46 A Cockburn and S Jones. Four Principles for Groupware Design. *Interacting with Computers: The Interdisciplinary Journal of Human-Computer Interaction*, 7(2):195–210. Elsevier. 1995.
- J47 HW Thimbleby, SP Marsh, SR Jones, and AG Cockburn. Trust in CSCW. In S Scrivener, editor, *Computer-Supported Cooperative Work*, pages 253–271. Ashgate Publishing, 1994.
- J48 H Thimbleby, A Cockburn, and S Jones. Hypercard: An Object-Oriented Disappointment. In P Gray and R Took, editors, *Building Interactive Systems: Architectures and Tools*, pages 35–55. Springer-Verlag, 1992.

Papers in Refereed Conference Proceedings

- C1 K Kargut, C Gutwin, A Cockburn. Effects of Device Environment and Information Layout on Spatial Memory and Performance in VR Selection Tasks. *Proceedings of ACM CHI 2024*. Hawai’i. 2024.
- C2 P Suwanaposee, C Gutwin, Z Chen, A Cockburn. ‘Specially For You’ – Examining the Barnum Effect’s Influence on the Perceived Quality of System Recommendations. *Proceedings of ACM CHI 2023*. Hamburg, Germany. 2023.
- C3 A Cockburn, P Quinn, C Gutwin, Z Chen, P Suwanaposee. Probability Weighting in Interactive Decisions: Evidence for Overuse of Bad Assistance, Underuse of Good Assistance. *Proceedings of ACM CHI 2022*. New Orleans, USA. 2022.
- C4 K Lam, C Gutwin, M Klarkowski, and A Cockburn. More Errors vs. Longer Commands: The Effects of Repetition and Reduced Expressiveness on Input Interpretation Error, Learning, and User Preference. *Proceedings of ACM CHI 2022*. New Orleans, USA. 2022.

- c5 A Goguet, C Gutwin, Z Chen, P Suwanaposee, and A Cockburn. Interaction Pace and User Preferences. *Proceedings of ACM CHI 2021*. Yokohama, Japan. 2021. **Honourable mention award**
- c6 K Lam, C Gutwin, M Klarkowski, and A Cockburn. The Effects of System Interpretation Errors on Learning New Input Mechanisms. *Proceedings of ACM CHI 2021*. Yokohama, Japan. 2021. **Honourable mention award**
- c7 S Liu, M Claypool, A Cockburn, R Eg, C Gutwin, and K Raaen. Datasets – Moving Target Selection with Delay. *Proceedings of the 12th ACM Multimedia Systems Conference (MMSys '21)*, Istanbul, Turkey. 2021.
- c8 N Bandeira Romao Tome, M Klarkowski, C Phillips, RL Mandryk, A Cockburn, and C Gutwin. Risking Treasure: Testing Loss Aversion in an Adventure Game. *Proceedings of ACM CHI PLAY 2020 Symposium on Computer-Human Interaction in Play*. Ottawa, Canada. 2020.
- c9 P Schmid, S Malacria, A Cockburn and M Nancel. Interaction Interferences: Implications of Last-Instant System State Changes. *Proceedings of ACM UIST'2020 Symposium on User Interface Software and Technology*. Minneapolis, USA. 2020.
- c10 P Palanque, A Cockburn and C Gutwin. A Classification of Faults Covering the Human-Computer Interaction Loop. *Proceedings of SafeComp 2020: 39th International Conference on Computer Safety, Reliability and Security*. Lisbon, Portugal. 2020.
- c11 A Mairena, M Dechant, C Gutwin and A Cockburn. A Baseline Study of Emphasis Effects in Information Visualization. *Proceedings of Graphics Interface 2020*. Toronto, Canada. 2020.
- c12 C Gutwin, A Cockburn, M van der Kamp, J Storrington and C Philips. Testing the Limits of the Spatial Approach: Comparing Retrieval and Revisitation Performance of Spatial and Paged Data Organizations for Large Item Sets. *Proceedings of Graphics Interface 2020*. Toronto, Canada. 2020.
- c13 A Cockburn, B Lewis, P Quinn and C Gutwin. Framing Effects Influence Interface Feature Decisions. *Proceedings of ACM CHI 2020*. Hawai'i. 2020.
- c14 B Lewis, G d'Eon, A Cockburn and D Vogel. KeyMap: Improving Keyboard Shortcut Vocabulary Using Norman's Mapping. *Proceedings of ACM CHI 2020*. Hawai'i. 2020.
- c15 A Goguet, S Malacria, A Cockburn, and C Gutwin. Reducing Error Aversion to Support Novice-to-Expert Transitions with FastTap. *Proceedings of the 31st Francophone Conference on Human-Computer Interaction – IHM 2019*. Grenoble, France. 2019. **Best paper award**
- c16 P Palanque, A Cockburn, L Désert-Legendre, C Gutwin and Y Délérès. Brace Touch: a Dependable, Turbulence-Tolerant, Multi-Touch Interaction Technique for Interactive Cockpits. *Proceedings of the 38th International Conference on Computer Safety, Reliability and Security - SafeComp 2019*. Turku, Finland. 2019.
- c17 C Martinie, P Palanque, E Bouzekri, A Cockburn, A Canny and E Barboni. Analysing and Demonstrating Tool-Supported Customizable Task Notations. *Proceedings of the 11th ACM SIGCHI Symposium on Engineering Interactive Computing Systems*. Valencia, Spain. 2019.

- c18 Q Roy, C Zakaria, W Kim, M Nancel, ST Perrault, A Misra and A Cockburn. A Comparative Study of Pointing Techniques for Eyewear Using a Simulated Pedestrian Environment. *Proceedings of the 17th IFIP TC.13 International Conference on Human-Computer Interaction*. Cyprus. 2019.
- c19 A Cockburn and C Gutwin. Anchoring Effects and Troublesome Asymmetric Transfer in Subjective Ratings. *Proceedings of ACM CHI 2019*. Glasgow, Scotland. 2019.
- c20 AF Mairena, C Gutwin, A Cockburn and LJ Elias The Effects of Feature Combination and Task Demands on Peripheral Notifications in Large Displays. *Proceedings of ACM CHI 2019*. Glasgow, Scotland. 2019.
- c21 M Claypool, A Cockburn and C Gutwin. Game Input with Delay – Moving Target Selection Parameters. *Proceedings of ACM Multimedia Systems Conference 2019*. Amhurst, MA. 2019.
- c22 A Cockburn, D Woolley, K Tran Pham Thai, D Clucas, S Hoermann, and C Gutwin. Reducing the Attentional Demands of In-Vehicle Touchscreens with Stencil Overlays. *Proceedings of ACM Automotive UI 2018*. Toronto, Canada. 2018.
- c23 A Cockburn, C Gutwin and A Dix. HARK No More: On the Preregistration of CHI Experiments. *Proceedings of ACM CHI 2018*. Montreal, Canada. 2018. Paper 141. **Best paper award**
- c24 A Goguet, G Casiez, A Cockburn and C Gutwin. Storyboard-Based Empirical Modelling of Touch Interface Performance. *Proceedings of ACM CHI 2018*. Montreal, Canada. 2018. Paper 445. **Honourable mention award**
- c25 A Cockburn, C Gutwin, P Palanque, Y Deleris, C Trask, A Coveney, M Yung and K MacLean. Turbulent Touch: Touchscreen Input for Cockpit Flight Displays. *Proceedings of ACM CHI 2017*. Denver, CO. 2017. 6742-6753.
- c26 C Gutwin, A Cockburn and N Gough. A Field Experiment of Spatially-Stable Overviews for Document Navigation. *Proceedings of ACM CHI 2017*. Denver, CO. 2017. 5905-5916.
- c27 C Gutwin, A Cockburn and A Coveney. Peripheral Popout: The Influence of Visual Angle on Popout Effects. *Proceedings of ACM CHI 2017*. Denver, CO. 2017. 208-219.
- c28 S Uddin, C Gutwin and A Cockburn. The Effects of Artificial Landmarks on Learning and Performance in Spatial-Memory Interfaces. *Proceedings of ACM CHI 2017*. Denver, CO. 2017. 3843-3855.
- c29 P Quinn and A Cockburn. When Bad Feels Good: Assistance Failures and Interface Preferences. *Proceedings of CHI'2016*. San Jose, California. 2016. 4005-4010.
- c30 B Lafreniere, C Gutwin, A Cockburn and T Grossman. Faster Command Selection on Touchscreen Watches *Proceedings of CHI'2016*. San Jose, California. 2016. 4663-4674.
- c31 K Schramm, C Gutwin and A Cockburn. Supporting Transitions to Expertise in Hidden Toolbars *Proceedings of CHI'2016*. San Jose, California. 2016. 4687-4698.
- c32 C Gutwin, C Rooke, A Cockburn, R Mandryk and B Lafreniere. Peak-End Effects on Player Experience in Casual Games *Proceedings of CHI'2016*. San Jose, California. 2016. 5608-5619. **Honourable mention award**

- c33 C Gutwin, A Cockburn and B Lafreniere. Testing the Rehearsal Hypothesis with Two FastTap Interfaces. *Proceedings of GI'2015*. Halifax, Canada. 2015. 223-231.
- c34 A Cockburn, C Gutwin and P Quinn. Examining the Peak-End Effects of Subjective Experience. *Proceedings of CHI'2015*. Seoul, S. Korea. 2015. 357-366.
- c35 S Malacria, J Acetuno, P Quinn, G Casiez, A Cockburn, and N Roussel. Push-Edge and Slide-Edge: Scrolling by Pushing Against the Viewport Edge. *Proceedings of CHI'2015*. Seoul, S. Korea. 2015. 2773-2776.
- c36 S Fitchett, A Cockburn and C Gutwin. Finder Highlights: Field Evaluation and Design of an Augmented File Browser. *Proceedings of CHI'2014*. Toronto, Canada. 2014. 3685-3694. **Honourable mention award**
- c37 C Gutwin, A Cockburn, J Scarr and S Malacria. Faster Command Selection on Tablets with FastTap. *Proceedings of CHI'2014*. Toronto, Canada. 2014. 2617-2626.
- c38 M Nancel and A Cockburn. Causality: A Conceptual Model of Interaction History. *Proceedings of CHI'2014*. Toronto, Canada. 2014. 1777-1786. **Honourable mention award**
- c39 J Scarr, A Cockburn, C Gutwin, A Bunt, and J Cechanowicz. The Usability of CommandMaps in Realistic Tasks. *Proceedings of CHI'2014*. Toronto, Canada. 2014. 2241-2250.
- c40 SA Seah, DM Plasencia, P Bennet, A Karnik, V Ortoloc, J Knibbe, A Cockburn, and S Subramanian. SensaBubble: A Chrono-Sensory Mid-Air Display of Sight and Smell. *Proceedings of CHI'2014*. Toronto, Canada. 2014. 2863-2872. **Honourable mention award**
- c41 P Quinn, S Malacria and A Cockburn. Touch Scrolling Transfer Functions. *Proceedings of ACM UIST'2013 Symposium on User Interface Software and Technology*. St Andrews, Scotland. 2013. 61-71.
- c42 S Malacria, J Scarr, A Cockburn, C Gutwin and T Grossman. Skillometers: Reflective Widgets that Motivate and Help Users to Improve Performance. *Proceedings of ACM UIST'2013 Symposium on User Interface Software and Technology*. St Andrews, Scotland. 2013. 321-330.
- c43 M Jain, A Cockburn, and S Madhvanath. Comparison of Phone-Based Distal Pointing Techniques for Point-Select Tasks. *Proceedings of INTERACT'13*. Cape Town, South Africa. 2013. 714-721.
- c44 S Malacria, G Bailly, J Harrison, A Cockburn, and C Gutwin. Promoting Hotkey Use through Rehearsal with ExposeHK. *Proceedings of CHI'2013*. Paris, France. 2013. 573-582.
- c45 J Scarr, A Cockburn, C Gutwin, and S Malacria. Testing the Robustness and Performance of Spatially Consistent Interfaces. *Proceedings of CHI'2013*. Paris, France. 2013. 3139-3148. **Honourable mention award**
- c46 S Fitchett, A Cockburn, and C Gutwin. Improving Navigation-Based File Retrieval. *Proceedings of CHI'2013*. Paris, France. 2013. 2329-2338. **Best paper award**
- c47 T Piumsomboon, A Clark, M Billinghamurst, and A Cockburn. User-Defined Gestures for Augmented Reality. *Proceedings of INTERACT'13*. Cape Town, South Africa. September. 2013. 282-299.

- c48 P Quinn, A Cockburn, G Casiez, N Roussel, and C Gutwin. Exposing and Understanding Scrolling Transfer Functions. *Proceedings of ACM UIST'2012 Symposium on User Interface Software and Technology*. Cambridge, Massachusetts, October. 2012. 341-350.
- c49 A Cockburn, P Quinn, C Gutwin, and S Fitchett. Improving Scrolling Devices with Document-Length-Dependent Gain. *Proceedings of CHI'2012*. Austin, Texas, May. 2012. 267-276.
- c50 J Scarr, A Cockburn, C Gutwin, and A Bunt. Improving Command Selection with CommandMaps. *Proceedings of CHI'2012*. Austin, Texas, May. 2012. 257-266. **Best paper award**
- c51 S Fitchett and A Cockburn. AccessRank: Predicting What Users Will Do Next. *Proceedings of CHI'2012*. Austin, Texas, May. 2012. 2239-2242. **Honourable mention award**
- c52 J Scarr, A Cockburn, C Gutwin, and P Quinn. Dips and Ceilings: Understanding and Supporting Transitions to Expertise in User Interfaces. *Proceedings of CHI'2011*. Vancouver, Canada, May. 2011. 2741-2750.
- c53 V Levesque, L Oram, KE MacLean, A Cockburn, N Marchuk, D Johnson, JE Colgate, and M Peshkin. Enhancing Physicality in Touch Interaction with Programmable Friction. *Proceedings of CHI'2011*. Vancouver, Canada, May. 2011. 2481-2490. **Best paper award**.
- c54 P Quinn, A Cockburn, KJ Raiha, and J Delemarche. On the Costs of Multiple Trajectory Pointing Methods. *Proceedings of CHI'2011*. Vancouver, Canada, May. 2011. 859-862. **Honourable mention award**
- c55 R Xiao, M Nacenta, RL Mandryk, A Cockburn and C Gutwin. Ubiquitous Cursor: A Comparison of Direct and Indirect Pointing Feedback in Multi-Display Environments. *Proceedings of Graphics Interface '2011*. St Johns, Canada. 2011. 135-142. **Michael A. J. Sweeney best student paper award**.
- c56 S Bateman, A Doucette, C Gutwin, RL Mandryk and A Cockburn. Effects of View, Input Device, and Track Width on Video Game Driving. *Proceedings of Graphics Interface '2011*. St Johns, Canada. 2011. 207-214.
- c57 S Tak, J Scarr, C Gutwin and A Cockburn. Supporting Window Switching With Spatially Consistent Thumbnails Zones: Design and Evaluation. *Proceedings of INTERACT'2011*. Lisbon, Portugal. 2011. 331-347.
- c58 B Ens, D Ahlstrom, A Cockburn and P Irani. Characterizing User Performance with Assisted Direct Off-Screen Pointing. *Proceedings of Mobile HCI'2011*. Stockholm. 2011. 331-347.
- c59 D Ahlstrom, A Cockburn, C Gutwin, and P Irani. Why it's Quick to be Square: Modelling New and Existing Hierarchical Menu Designs. *Proceedings of CHI'2010*. Atlanta, GA, April. 2010. 1371-1380. **Best paper nomination**.
- c60 J Oosterman and A Cockburn. An Empirical Comparison of Tag Clouds and Tables. *Proceedings of OzCHI'2010*. Brisbane, Australia, 22-26 November. 2010. 288-295.
- c61 S Fitchett and A Cockburn. Using Multitouch Input to Support Hybrid Relative and Absolute Mobile Scrolling *People and Computers 24*. *Proceedings of the British HCI Conference*. Dundee, 6-10 September. 2010.

- c62 S Fitchett and A Cockburn. Evaluating Flick-Scrolling and Tilt Scrolling for Reading Tasks on Mobile Devices. *Proceedings of OzCHI'2009*. Melbourne, Australia, 23-27 November. 2009. 225-232.
- c63 P Quinn and A Cockburn. Zoofing! Faster List Selections with Pressure-Zoom-Flick-Scrolling. *Proceedings of OzCHI'2009*. Melbourne, Australia, 23-27 November. 2009. 185-192.
- c64 S Tak and A Cockburn. Window Watcher: A Visualisation Tool for Understanding Window Switching Activities. *Proceedings of OzCHI'2009*. Melbourne, Australia, 23-27 November. 2009. 241-248.
- c65 S Tak, A Cockburn, K Humm, D Ahlstrom, C Gutwin and J Scarr. Improving Window Switching Interfaces. *Proceedings of INTERACT'2009*. Uppsala, Sweden, 24-28 August. 2009. 187-200.
- c66 J Zucco, B Thomas, K Grimmer-Sommers and A Cockburn. A Comparison of Menu Configurations and Pointing Devices for use with Wearable Computers while Mobile and Stationary. *Proceedings of ACM ISWC'2009: 13th International Symposium on Wearable Computers*. Linz, Austria, 5-7 September. 2009. 63-70.
- c67 J Alexander, A Cockburn, S Fitchett, C Gutwin and S Greenberg. Revisiting Read Wear: Analysis, Design, and Evaluation of a Footprints Scrollbar. *Proceedings of ACM CHI'2009 Conference on Human Factors in Computing Systems*. Boston, MA, 5 April - 9 April. 2009. 1665-1674.
- c68 E Tanvir, J Cullen, P Irani and A Cockburn. Improving Selection in Cascading Pull-Down Menus. *Proceedings of ACM CHI'2008 Conference on Human Factors in Computing Systems*. Florence, Italy, 5 April - 10 April. 2008. 1381-1384.
- c69 J Alexander and A Cockburn. An Empirical Characterisation of Electronic Document Navigation. *Proceedings of Graphics Interface '08*. Windsor, Ontario, Canada, May 28-30. 2008. 123-130. **Michael A. J. Sweeney best student paper award.**
- c70 P Quinn, A Cockburn, Indratmo and C Gutwin. An Investigation of Dynamic Landmarking Functions. *Proceedings of Advanced Visual Interfaces (AVI'08)*. Napoli, Italy, May 28-30. 2008. ACM Press.
- c71 P Quinn and A Cockburn. The Effects of Menu Parallelism on Visual Search and Selection. *Proceedings of AUIC'08: Australasian User Interface Conference*. 2008, 79-84. ACM Press.
- c72 V Buchmann, M Billinghamurst and A Cockburn. Directional Interfaces for Wearable Augmented Reality. *Proceedings of CHINZ'08 2008*, 47-54. ACM Press.
- c73 J Looser, M Billinghamurst, R Grasset and A Cockburn. An Evaluation of Virtual Lenses for Object Selection in Augmented Reality. *Proceedings of GRAPHITE 2007*. Perth, Australia, 1-4 December. 2007, 203-210. ACM Press.
- c74 A Cockburn, C Gutwin and S Greenberg. A Predictive Model of Menu Performance *Proceedings of ACM CHI'2007 Conference on Human Factors in Computing Systems* San Jose, CA, 24-27 April. 2007, 627-636. ACM Press.
- c75 A Cockburn, PO Kristennson, J Alexander and S Zhai. Hard Lessons: Effort-Inducing Interfaces Benefit Spatial Learning *Proceedings of ACM CHI'2007 Conference on Human Factors in Computing Systems* San Jose, CA, 24-27 April. 2007, 1571-1580. ACM Press.

- c76 M Hancock, S Carpendale and A Cockburn. Shallow Depth 3D Interaction: Design and Evaluation of One-, Two-, and Three-Touch Techniques *Proceedings of ACM CHI'2007 Conference on Human Factors in Computing Systems* San Jose, CA, 24–27 April. 2007, 1147–1156. **Nominated for best paper award.** ACM Press.
- c77 G Ramos, A Cockburn, M Beaudouin-Lafon and R Balakrishnan. Pointing Lenses *Proceedings of ACM CHI'2007 Conference on Human Factors in Computing Systems* San Jose, CA, 24–27 April. 2007, 757–766.. ACM Press.
- c78 T Bell, A Cockburn, A Wingkvist and R Green. Podcasts as a supplement in tertiary education: an experiment with two Computer Science courses *Proceedings of MoLTA 2007: Conference on Mobile Learning Technologies and Applications* Auckland, New Zealand. 2007, 70–77.
- c79 T Amer, A Cockburn, R Green and G Odgers. Evaluating Swiftpoint as a Mobile Device for Direct Manipulation Input *Proceedings of AUIC'2007 Australasian User Interface Conference* Ballarat, Australia. 2007, 63–70.
- c80 J Hauber, H Regenbrecht, M Billinghamurst and A Cockburn. Spatiality in Videoconferencing: Trade-offs between Efficiency and Social Presence *Proceedings of ACM CSCW'2006 Conference on Computer Supported Cooperative Work* Banff, Canada, November 4–8. 2006, 413–422.
- c81 V Jurgens, A Cockburn and M Billinghamurst Depth Cues For Augmented Reality Stakeout *Proceedings of CHINZ'2006: Conference of the New Zealand Chapter of the ACM Special Interest Group on Human-Computer Interaction* Canterbury, New Zealand 2006, 117–124.
- c82 C Gutwin and A Cockburn. Improving List Revisitation with LispMaps *Proceedings of ACM AVI'2006 Conference on Advanced Visual Interfaces*. Venice, Italy, May 23–26. 2006, 396–403. 32 accepted from 127 submissions.
- c83 A Cockburn and A Gin. Faster Cascading Menu Selections with Enlarged Activation Areas *Proceedings of Graphics Interface 2006*. Quebec City, Canada, June 7–9. 2006, 65–71.
- c84 A Cockburn and P Brock. Human On-Line Response to Visual and Motor Target Expansion *Proceedings of Graphics Interface 2006*. Quebec City, Canada, June 7–9. 2006, 81–87.
- c85 A Cockburn, C Gutwin and J Alexander. Faster Document Navigation with Space-Filling Thumbnails *Proceedings of ACM CHI'2006 Conference on Human Factors in Computing Systems* Montreal, Canada, 24–27 April. 2006, 1–10. ACM Press.
- c86 J Looser, A Cockburn and J Savage. On the Validity of Using First-Person Shooters for Fitts' Law Studies Acceptance rate: 35%, 26 from 73. *People and Computers XVII (Volume 2): British Computer Society Conference on Human Computer Interaction* Edinburgh, Scotland, September 2005, 33–36. Springer-Verlag.
- c87 J Savage and A Cockburn. Comparing Automatic and Manual Zooming Methods for Acquiring Off-Screen Targets *People and Computers XIX: British Computer Society Conference on Human Computer Interaction* Edinburgh, Scotland, September 2005, 439–454. Springer-Verlag. Acceptance rate: 34%, 31 from 92.
- c88 J Hauber, H Regenbrecht, A Hills, A Cockburn and M Billinghamurst. Social Presence in Two- and Three-dimensional Videoconferencing *Proceedings of 8th Annual International Workshop on Presence* London, UK, September 21–23, 2005, 189–198.

- c89 A Cockburn, J Savage and A Wallace. Tuning and Testing Scrolling Interfaces that Automatically Zoom *Proceedings of ACM CHI'2005 Conference on Human Factors in Computing Systems* Portland, Oregon, 2–7 April. 2005, 71–80. ACM Press. Acceptance rate: 25%, 93 from 371. **Nominated for best paper award.**
- c90 T Wright and A Cockburn. Evaluation of Two Textual Programming Notations for Children *Proceedings of the Sixth Australasian User Interface Conference (AUIC2005)* Newcastle, Australia, 2–3 February 2005, 55–62. Australian Computer Society.
- c91 L Rennie and A Cockburn. Aiding Text Entry of Foreign Alphabets with Visual Keyboard Plus *Proceedings of the Sixth Australasian User Interface Conference (AUIC2005)* Newcastle, Australia, 2–3 February 2005, 119–125. Australian Computer Society.
- c92 S Blinmann and A Cockburn. Program Comprehension: Investigating the effects of naming style and documentation *Proceedings of the Sixth Australasian User Interface Conference (AUIC2005)* Newcastle, Australia, 2–3 February 2005, 73–78. Australian Computer Society.
- c93 J Looser, M Billinghamurst and A Cockburn. Through the Looking Glass: The Use of Lenses as an Interface Tool for Augmented Reality *Proceedings of Graphite'04: International Conference on Computer Graphics and Interactive Techniques* Singapore, 15–18 June 2004, 204–211.
- c94 V Buchmann, S Violich, M Billinghamurst and A Cockburn. FingARtips — Gesture Based Direct Manipulation in Augmented Reality *Proceedings of Graphite'04: International Conference on Computer Graphics and Interactive Techniques* Singapore, 15–18 June 2004, pp 212–221.
- c95 A Cockburn. Revisiting 2D vs 3D Implications on Spatial Memory *Proceedings of the Fifth Australasian User Interface Conference (AUIC2004)* Dunedin, New Zealand, 18–22 January 2004, pp 25–32. Australian Computer Society.
- c96 A Wallace, J Savage and A Cockburn. Rapid Visual Flow: How Fast is Too Fast? *Proceedings of the Fifth Australasian User Interface Conference (AUIC2004)* Dunedin, New Zealand, 18–22 January 2004, pp 117–122. Australian Computer Society.
- c97 A Cockburn, J Looser and J Savage. Around the World in Seconds with Speed-Dependent Automatic Zooming *Paper and Video in the Conference Supplement of the ACM Conference on User Interface Software and Technology* Vancouver, Canada, November 2003, pp 35–36.
- c98 A Cockburn and J Savage. Comparing Speed-Dependent Automatic Zooming with Traditional Scroll, Pan and Zoom Methods *People and Computers XVII: British Computer Society Conference on Human Computer Interaction* Bath, England, September 2003, pages 87–102. Springer-Verlag.
- c99 A Cockburn and A Firth. Improving the Acquisition of Small Targets *People and Computers XVII: British Computer Society Conference on Human Computer Interaction* Bath, England, September 2003, pages 181–196. Springer-Verlag.
- c100 A Cockburn and A Siresena. Evaluating Mobile Text Entry with the Fastap Keypad Acceptance rate: 33%, 8 from 24. *People and Computers XVII (Volume 2): British Computer Society Conference on Human Computer Interaction* Bath, England, September 2003, pages 77–80. Springer-Verlag.

- C101 T Wright and A Cockburn. A Language and Task-based Taxonomy of Programming Environments *Proceedings of IEEE Symposium on Visual Languages and Formal Methods*. Auckland, October 28–31 2003, pages 192–194. Acceptance rate: 33%, 8 from 24.
- C102 T Wright and A Cockburn. An Evaluation of the Effects of Different Forms of Computer Supported Collaboration on Problem Solving Strategies *Proceedings of CHINZ'03. Annual conference of the New Zealand Chapter of the ACM Special Interest Group on Human-Computer Interaction*. Otago, July 2003, pages 99–104. Acceptance rate: 59%, 19 from 32.
- C103 T Wright and A Cockburn. A Framework for Evaluating Multiple Language Programming Environments *Proceedings of CHINZ'03. Annual conference of the New Zealand Chapter of the ACM Special Interest Group on Human-Computer Interaction*. Otago, July 2003, pages 21–26. Acceptance rate: 59%, 19 from 32.
- C104 M Apperley, P Carter, C Churcher, A Cockburn, M Jones, B Lobb, K Novins, C Phillips and W Wong. State of the Art: HCI in New Zealand (HCI Societies Worldwide Review) *Proceedings of INTERACT'03: Ninth IFIP TC13 International Conference on Human-Computer Interaction* Zurich, Switzerland, Sept 1–5 2003
- C105 M Moyle and A Cockburn. The Design and Evaluation of a Flick Gesture for ‘Back’ and ‘Forward’ in Web Browsers *Proceedings of the Fourth Australasian User Interface Conference (AUIC2003)* Adelaide, Australia, 4–7 February 2003, pages 39–46.
- C106 M JasonSmith and A Cockburn. Get a Way Back: An Evaluation of Temporal Lists *Proceedings of the Fourth Australasian User Interface Conference (AUIC2003)* Adelaide, Australia, 4–7 February 2003, pages 33–38.
- C107 T Wright and A Cockburn. Mulspreen: A MULTiple Language Simulation PROgramming ENvironment *Proceedings of the 2002 IEEE Symposia on Human Centric Computing Languages and Environments* Arlington, VA, Sept 3–6 2002, pages 101–103.
- C108 T Wright and A Cockburn. Evaluating Computer-Supported Collaboration for Learning a Problem Solving Task, *Proceedings of the International Conference on Computers in Education*, Auckland, NZ 2002, pages 266–267.
- C109 M Moyle and A Cockburn. Analysing Mouse and Pen Flick Gestures *Proceedings of the SIGCHI-NZ Symposium On Computer-Human Interaction* Hamilton, New Zealand, 11–12 July. 2002, pages 19–24.
- C110 A Cockburn and B McKenzie. Evaluating the Effectiveness of Spatial Memory in 2D and 3D Physical and Virtual Environments *Proceedings of ACM CHI'2002 Conference on Human Factors in Computing Systems* Minneapolis, Minnesota, 20–25 April. 2002, pages 203–210. Addison-Wesley. Acceptance rate: 15%, 61 from 409.
- C111 A Cockburn, B McKenzie, and M JasonSmith. Evaluating a Temporal Behaviour for Web Browsers’ Back and Forward Buttons *CD and on-line Proceedings of WWW2002, the 11th International World Wide Web Conference* Honolulu, May 7–11. 2002. www2002.org
- C112 M Moyle and A Cockburn. Gesture Navigation: An Alternative Back for the Future *Extended Abstracts of ACM CHI'2002 Conference on Human Factors in Computing Systems* Minneapolis, Minnesota, 20–25 April. 2002, pages 822–823. ACM Press. Acceptance rate: 17%, 62 from 374.

- C113 L Butts and A Cockburn. An Evaluation of Mobile Phone Text Input Methods *Proceedings of Australian User Interface Conference*. 2002, pages 55–59. Australian Computer Society Inc.
- C114 T Wright and A Cockburn. Solo, Together, Apart: Evaluating Modes of CSCL for Learning a Problem Solving Task *Proceedings of ACM Computer Supported Collaborative Learning (CSCL) Conference 2002*. newmedia.colorado.edu/cscl/06.pdf
- C115 T Bell, A Cockburn, B McKenzie and J Vargo. Flexible Delivery Damaging to Learning? Lessons from the Canterbury Digital Lectures Project *Proceedings of ED-MEDIA'2001: World Conference on Educational Multimedia, Hypermedia and Telecommunications* Tampere, Finland, June 25–30. 2001, pages 117–122. Association for the Advancement of Computing in Education.
- C116 A Cockburn and B McKenzie. 3D or Not 3D? Evaluating the Effect of the Third Dimension in a Document Management System *Proceedings of ACM CHI'2001 Conference on Human Factors in Computing Systems* Seattle, Washington, March 31–April 6. 2001, pages 434–441. Addison-Wesley. Acceptance rate: 19%, 69 from 353.
- C117 B McKenzie and A Cockburn. An Empirical Analysis of Web Page Revisitation *34th Hawaiian International Conference on System Sciences, HICSS34 2001*. IEEE Computer Society Press.
- C118 T Wright and A Cockburn. Writing, Reading, Watching: A Task-Based Analysis and Review of Learners' Programming Environments *International Workshop on Advanced Learning Technologies, IWALT 2000*, pages 167–170. IEEE Computer Society Press.
- C119 A Cockburn and B McKenzie. An Evaluation of Cone Trees *People and Computers XIV: British Computer Society Conference on Human Computer Interaction 2000*, pages 425–436. Springer-Verlag. Acceptance rate: 27 from 57.
- C120 A Cockburn and S Greenberg. Issues of Page Representation and Organisation in Web Browser's Revisitation Tools *OzCHI'99: Australian Conference on Computer-Human Interaction*, Wagga Wagga, November 28–30, 1999, pages 7–14. Acceptance rate: 50%.
- C121 A Cockburn, S Greenberg, B McKenzie, M JasonSmith and S Kaasten. WebView: A Graphical Aid for Revisiting Web Pages *OzCHI'99: Australian Conference on Computer-Human Interaction*, Wagga Wagga, November 28–30, 1999, pages 15–22. Acceptance rate: 50%.
- C122 S Greenberg and A Cockburn. Getting Back to Back: Alternate Behaviors for a Web Browser's Back Button *5th Conference on Human Factors and the Web, Gaithersburg, Maryland. June 3, 1999*. Acceptance rate: 11 from 27.
- C123 P Weir and A Cockburn. Distortion-Oriented Workspace Awareness in DOME *British Computer Society Conference on Human-Computer Interaction. Imperial College, London. 1–4 September, 1998.*, pages 239–252. Springer-Verlag. Acceptance rate: 21 from 89.
- C124 E Ng, T Bell and A Cockburn. Improvements to a Pen-Based Musical Input System *OzCHI'98: The Australasian Conference on Computer-Human Interaction. Adelaide, Australia. 29th November to 4th December, 1998.* 1998., pages 239–252. IEEE Computer Society Press.
- C125 A Cockburn and T Bell. Extending HCI in the Computer Science Curriculum *ACM Australasian Computer Science Education Conference ACSE'98. Brisbane, Australia, 8–10 July, 1998*, pages 113–120. ACM Press. Acceptance rate: 28 from 59.

- C126 A Cockburn and A Bryant. Cleogo: Collaborative and Multi-Paradigm Programming for Kids *APCHI'98: Asia Pacific Conference on Computer Human Interaction*. Japan. July 15–17, 1998, pages 187–192. IEEE Computer Society Press.
- C127 N Churcher, A Cockburn, B McMaster, J Horlor. CUTE—The Design and Evolution of a First Year Programming Environment *Software Engineering: Education and Practice '98*. Dunedin. January, 1998, pages 142–148. IEEE Computer Society Press.
- C128 A Cockburn and T Dale. CEVA: A Tool for Collaborative Video Analysis *ACM Group'97: International Conference on Supporting Group Work* Phoenix, Arizona, November 16–19, 1997, pages 47–55. ACM Press. Acceptance rate: 44 from 104.
- C129 A Cockburn and N Churcher. Towards Literate Tools for Novice Programmers *ACM Australasian Computer Science Education Conference ACSE'97*. Melbourne, Australia, 2-4 July 1997, pages 163–169. ACM Press. Acceptance rate: 31 from 46.
- C130 N Churcher and A Cockburn. An Immersion Model for Software Engineering Projects *ACM Australasian Computer Science Education Conference ACSE'97*. Melbourne, Australia, 2-4 July 1997, pages 107–116. ACM Press. Acceptance rate: 31 from 46.
- C131 A Cockburn and S Jones. Design Issues for World Wide Web Navigation Visualisation Tools *RIA0'97: The Fifth Conference on Computer-Assisted Research of Information*. McGill University, Montreal, Quebec, Canada, 25th-27th June 1997. pages 55–74. Acceptance rate: 35 from 90.
- C132 J Anstice, T Bell, A Cockburn and M Setchell. The Design of a Pen-Based Musical Input System. *OzCHI'96: The Sixth Australian Conference on Computer-Human Interaction*, Hamilton, New Zealand, November 24–27, 1996, pages 260–267. IEEE Press. Acceptance rate: 38 from 79.
- C133 A Cockburn and A Bryant. Do It This Way: Equal Opportunity Programming for Kids *OzCHI'96: The Sixth Australian Conference on Computer-Human Interaction*, Hamilton, New Zealand, November 24–27, 1996, pages 246–251. IEEE Press. Acceptance rate: 38 from 79.
- C134 A Tay and A Cockburn User Interfaces for Workflow Systems: Designing for End-User Tailorability Short paper in *OzCHI'96: The Sixth Australian Conference on Computer-Human Interaction*, Hamilton, New Zealand, November 24–27, 1996, pages 301–302. IEEE Press.
- C135 A Cockburn. Collaborative Usability Analysis Using Video Protocols *APCHI'96: The First Asia Pacific Conference Human Computer Interaction*, Information Technology Institute Singapore, 25–28 June, 1996, pages 67–80. Acceptance rate: lower than 50%.
- C136 S Greenberg, C Gutwin, and A Cockburn. Using Distortion-Oriented Displays to Support Workspace Awareness. *British Computer Society Conference on Human-Computer Interaction. Imperial College, London. 20–23 August, 1996*. Cambridge University Press. Acceptance rate: 23 from 63.
- C137 A Cockburn and S Greenberg. Children's Collaboration Styles in a Newtonian Microworld. In *Companion Proceedings of CHI'96. ACM Conference on Human Factors in Computing Systems* Vancouver, April 13–18 1996, pages 181–182. ACM Press. Acceptance rate: 78 from 335.
- C138 S Greenberg, C Gutwin, and A Cockburn. Awareness Through Fisheye Views in Relaxed-WYSIWIS Groupware *Proceedings of Graphics Interface, May 21–24, 1996. Toronto*, pages 28–38. Morgan-Kauffman. Acceptance rate: 29 from 73.

- C139 SRA Jones and AJG Cockburn. A Study of Navigational Support Provided by Two World Wide Web Browsing Applications. In *Hypertext '96. Seventh ACM Conference on Hypertext. Washington DC March 16–20, 1996*, pages 161–169. ACM Press. Acceptance rate: 24 from 85.
- C140 AJG Cockburn and SRA Jones. Trails, Trials, and Tribulations: Unravelling Navigational Problems in the World Wide Web. In *Proceedings of the 5th Workshop on Information Technologies and Systems (WITS '95). Nijenrode University, The Netherlands. 9–10 December.*, pages 46–55, 1995. Acceptance rate: 24 from 70.
- C141 A Cockburn and S Greenberg. TurboTurtle: A Collaborative Microworld for Exploring Newtonian Physics. In *ACM Conference on Computer Supported Cooperative Learning (CSCL '95). Bloomington, Indiana. October 17–20, 1995*, pages 62–66. Lawrence Erlbaum Associates, Inc, October 1995.
- C142 AJG Cockburn. TurboTurtle: Educating Children in an Alternative Reality Universe. In *Proceedings of the 1994 Computer Human Interaction Specialist Interest Group of the Ergonomics Society of Australia (OZCHI'94)*. November 28 to December 1. Melbourne., pages 99–105, 1994.
- C143 A Cockburn and S Jones. Four Principles for Groupware Design. In *Proceedings of the 1994 Computer Human Interaction Specialist Interest Group of the Ergonomics Society of Australia (OZCHI'94)*. November 28 to December 1. Melbourne., pages 21–26, 1994.
- C144 AJG Cockburn and S Greenberg. Making Contact: Getting the Group Communicating with Groupware. In *COOCS '93. The ACM Conference on Organisational Computing Systems. November 1st to 4th. Milpitas, California.*, pages 31–41, 1993.
- C145 AJG Cockburn. Supporting Social Awareness in Distributed Work. In P *et al* Collings, editor, *Proceedings of OZCHI 1993. CHISIG Annual Conference*. Canberra. November., pages 224–241, 1993.
- C146 AJG Cockburn and HW Thimbleby. Reducing User Effort in Collaboration Support. In WD Gray, WE Hefley, and D Murray, editors, *Proceedings of the 1993 International Workshop on Intelligent User Interfaces, Orlando, Florida. January 4–7.*, pages 215–218. New York: ACM Press, January 1993.
- C147 AJG Cockburn and HW Thimbleby. Automatic Conversational Context: Avoiding Dependency on User Effort in Groupware. In MJ Rees and R Iannella, editors, *Interface Technology: Advancing Human-Computer Communication. Proceedings of OZCHI 1992. CHISIG Annual Conference*. Bond University Australia, Queensland. November 26–27, pages 142–149, 1992.
- C148 AJG Cockburn. Reflexive CSCW and Managing Commitments. In *IEEE Colloquium on CSCW: Some Fundamental Issues*. London, March 15, number 065 in 1991, pages 9/1–9/5. IEE, 1991.

Patents and Patent Applications

- P1 Andy Cockburn, Mark Billingham, Dinesh Mandalapu. User Interface Device. US Patent Application Number: PCT/IN2011/000897.

Other: Refereed Videos, Media Reports and Non-Refereed Publications

- 01 L Besanon, Y Jansen, A Cockburn, and P Dragicevic. Definitely Maybe: Hedges And Boosters in the HCI Literature. OSF Preprints. <https://osf.io/mjg7h/>, 2021.
- 02 National Science Challenges report. ‘Meet the researchers: Andy Cockburn, Turbulent touch’. 22/03/2019.
- 03 Marsden update. ‘Young researchers experience international science.’ (PhD student Stephen Fitchett). February 2012.
- 04 Magazine article. New Zealand Listener. “Charge of the tech brigade”. Mar 31, 2012.
- 05 Newspaper article. The Press. “Conference acceptance a coup”. Dec 22, 2011.
- 06 Newspaper article. The Press. “Intuitive Innovations”. Feb 22, 2011.
- 07 UC Communications. ‘UC/Microsoft internship opens up world for computer science student.’ Mar 22, 2011.
- 08 Marsden update. ‘Highlights of the 2010 funding round.’ November 2010.
- 09 Magazine article. ‘Profiling Excellence’. Royal Society of New Zealand. ‘Thirty years of research in a single hit’, March 2010.
- 010 Newspaper article. Sydney Morning Herald. ‘Computer mouse still rules, says expert’, Jan 19, 2010. (Widely syndicated through AAP.)
<http://news.smh.com.au/breaking-news-national/computer-mouse-still-rules-says-expert-20100119-mhnp.html>
- 011 Newspaper article. Dominion Post. ‘Next-step mouse on its way’, Feb 1, 2010.
- 012 Radio Interview (Live). ABC Perth. ‘Drive Time’ program. January 19th, 2010.
- 013 Newspaper article. The Press. ‘Boost for Computer Activation Project’, Nov. 5, 2009.
- 014 Radio interview (Sweden). ‘Highlights of the INTERACT Conference’, Sept 3 2009. <http://www.sr.se/cgi-bin/p1/program/artikel.asp?ProgramID=406&nyheter=1&artikel=3075930>
- 015 Newspaper article. Dominion Post. ‘Kiwi mouse comes out of hiding’, March 10, 2008. <http://www.stuff.co.nz/4433363a28.html>
- 016 Newspaper article. Christchurch Press. ‘Fast Kiwi mouse attracts interest from big players’, March 5, 2008. <http://www.stuff.co.nz/4425499a28.html>
- 017 Newspaper article. The Age (Australia). ‘Kiwi mouse attracts interest’, March 5, 2008. <http://www.theage.com.au/news/biztech/kiwi-mouse-attracts-interest/2008/03/10/1204998342852.html>
- 018 Magazine article. Marsden Update. ‘Document Navigation’, December 2008.
- 019 Interview. IEEE Spectrum. ‘A New Spin on the Computer Mouse’, October 2006. <http://www.spectrum.ieee.org/oct06/4668>

- 020 Video. Faster Document Navigation with Space-Filling Thumbnails *Video in the Video Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI2006)* Montreal, Canada, April 2006. <http://www.cosc.canterbury.ac.nz/~andy/sft.mov>
- 021 Video. Tuning and Testing Scrolling Interfaces that Automatically Zoom *Video in the Refereed Video Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI2005)* Portland, Oregon, April 2005. <http://www.cosc.canterbury.ac.nz/~andy/tuning-n-testing.wmv>
- 022 Video. Around the World in Seconds with Speed-Dependent Automatic Zooming *Video and Demonstration at the ACM Conference on User Interface Software and Technology* Vancouver, Canada, November 2003. <http://www.cosc.canterbury.ac.nz/~andy/uist-sdaz.avi>
- 023 Radio Interview. BBC World Service. 'Go Digital' program. January 14th, 2003 <http://news.bbc.co.uk/2/hi/technology/2677337.stm>
- 024 Nature Journal's Science Update. 'Browsers go back to the future' www.nature.com/nsu/021230/021230-3.html
- 025 'The Register' in the UK. 'Kiwis redesign the Back button'. <http://www.theregister.co.uk/content/6/28729.html>
- 026 Canadian Broadcast Corporation. Programmers try out alternative to browser's back button. http://cbc.ca/storyview/CBC/2002/12/30/back_button021230
- 027 Slashdot. 'Redesigning The "Back" Button' slashdot.org/article.pl?sid=02/12/30/1727206&mode=thread&tid=95
- 028 NZOOM's Technology Page. technology.nzzoom.com/technology_subcategory/0,1610,113-116,00.html
- 029 Newspaper Interview. Christchurch Press. 'Better Going Back', Feb 11, 2003.
- 030 University Chronicle. "Taking the frustration out of the web", Feb 20, 2003.
- 031 S Greenberg, C Gutwin, and A Cockburn. Applying Distortion-Oriented Displays to Groupware. Video Programme of the ACM Conference on Computer Supported Cooperative Work. 1996.
- 032 S Greenberg, C Gutwin, and A Cockburn. Sharing fisheye views in relaxed-WYSIWIS groupware applications. Technical Report 95/577/29, Department of Computer Science, University of Calgary, Calgary, CANADA., 1996.
- 033 S Greenberg, C Gutwin, and A Cockburn. Using distortion-oriented displays to support workspace awareness. Technical Report 96/581/01, Department of Computer Science, University of Calgary, Calgary, CANADA., 1996.
- 034 S Greenberg, C Gutwin, M Roseman, and A Cockburn. From awareness to teamrooms, groupweb and turboturtle: Eight snapshots of recent work in the grouplab project. Technical Report 95/580/32, Department of Computing Science, University of Calgary, Canada, December 1995.
- 035 A Cockburn and S Greenberg. The design and evolution of turboturtle: A collaborative microworld for exploring newtonian physics. Technical Report tr-cosc.03.95, University of Canterbury, Christchurch. New Zealand, 1995.

- o36 AJG Cockburn. Social awareness without mega-bits. In *Proceedings of Uniforum. Rotoroa, New Zealand.*, 1994.
- o37 AJG Cockburn. *Groupware design: principles, prototypes, and systems*. PhD thesis, Department of Computing Science and Mathematics, University of Stirling, Scotland. FK9 4LA, 1993. Available from British Libraries, or as Technical Report 107 from the University of Stirling.
- o38 SRA Jones and AJG Cockburn. Reducing requirements in computer supported writing tasks, 1993. Paper presented at: 6th UK conference on computers and writing. University of Wales. April 13-15.
- o39 AJG Cockburn and S Greenberg. Making contact: getting the group communicating. Technical Report 93/498/03, Department of Computer Science, University of Calgary, Alberta T2N 1N4 Canada, 1993.
- o40 AJG Cockburn and HW Thimbleby. Reducing user effort in collaboration support. Technical Report 93, University of Stirling, Stirling, Scotland, FK9 4LA, 1992.
- o41 AJG Cockburn and HW Thimbleby. A reflexive perspective of CSCW. *ACM SIGCHI Bulletin*, 23(3):63–68, July 1991.
- o42 AJG Cockburn. A user's guide to TELEFREEK: a communication and social awareness environment. Technical Report 105, Department of Computing Science and Mathematics, University of Stirling, Stirling, Scotland, FK9 4LA, 1993.
- o43 AJG Cockburn. A user's guide to *mona*: an enhanced email system. Technical Report 97, Department of Computing Science and Mathematics, University of Stirling, Stirling, Scotland, FK9 4LA, 1992.

Grants and industry research/consulting

Project	Source	Date	Amount
Understanding the human experience of intelligent user interfaces	Marsden. (19-UOC-021)	2020-2023	\$530,000
Travel expenses	Airbus & U. Paul Sabatier	2018	≈ \$7,000
In-Vehicle Touchscreens	NZ Science for Innovation	2017	\$200,000
Travel expenses	ACM CHI Steering Committee	2017	≈ \$4,000
Travel expenses	Australian National University	2016	≈ \$3,000
Travel expenses	Airbus & U. Paul Sabatier	2016	≈ \$5,000
Travel expenses	ACM CHI Academy Award (Seoul)	2015	≈ \$5,000
Benjamin Meaker Visiting Professorship	University of Bristol	2013	£7,000
Understanding and improving the novice to expert transition with user interfaces	Marsden. (10-UOC-020)	2011-2014	\$496,000
Dynamic Multimodal Affordances	HP Innovations Award	2011-2012	US\$40,000
Travel expenses	ACM UIST conference	June 2009	≈ \$4,000
Postdoctoral Fellow	BuildIT	2011-2012	\$116,000
Dynamic Multimodal Affordances	HP Innovations Award	2010-2011	US\$40,000
Travel expenses	ACM UIST conference	June 2009	≈ \$4,000
Scrolling device studies	Logitech Inc	2010	\$10,000
Canadian GRAND project	Canadian Centres of Excellence	2010-2014	C\$25,000,000
BuildIT Emerging Researcher Mentoring Fund	BuildIT	2010	\$8,000
Pythodogical	UC Teaching and Learning	2009-2010	\$15,000
Dynamic Multimodal Affordances	HP Innovations Award	2009-2010	US\$40,000
Travel expenses	ACM UIST conference	June 2009	≈ \$4,000
Travel expenses	ACM CHI conference	Nov. 2008	≈ \$4,000
Improving document navigation interfaces	Marsden. (07-UOC-013)	2007–2010	\$400,000
Accessible AR on Everyday Devices	FRST (UOCX0704)	2007–2011	\$1,875,000
Usability training	Jade Software	2007	≈ \$2,000
Usability training	AMI	2007	≈ \$2,000
Expenses	Logitech Inc., California	April 2007	≈ \$5,000
Expenses	Logitech Inc., California	July 2006	≈ \$2,000
Grant in aid	University of Toronto	July 2006	≈ \$8,000
Grant in aid	University of Saskatoon	June 2006	≈ \$6,000
Grant in aid	University of Calgary	May 2006	≈ \$3,000
Grant in aid	IBM Almaden Research, California	April 2006	≈ \$3,000
NOKIA Pocket (Assoc. with HIT Lab).	Nokia Research Center	2006.	≈ \$100,000.
Speed-Dependent Automatic Zooming	Computer Science	2003	\$3,000
Evaluation of Speed-Coupled Scrolling	Computer Science	2002	\$3,000
Multimedia Learning of Graphics (Assoc.)	UoC Teaching and Learning Ctte.	2002	\$5,000
Fastap Evaluation	Digit Wireless LLC, MA, USA.	2002	\$3,000
Supermedia—The MagicBook Project (Assoc.)	NZ NERF (UOCX0204)	2002	\$333,000
Practical Flexible Delivery of Virtual Learning	UoC (U6360)	1999–2002	\$40,000
Collaborative Programming for Children	Marsden, Royal Soc. (UOC805)	2000–2002	\$135,000
Evaluation of Back/Forward in Web-browsers	Computer Science	2001	\$3,000
Interfaces for WWW navigation.	Microsoft Research, USA.	1999	≈ \$8,000
HCI and multi-media research lab	Computer Science	1999	\$7,400
Video analysis and multi-media lab	UoC (2313728)	1996	\$30,000
Collaborative video analysis	Computer Science	1996	\$3,000
Collaborative microworlds	Computer Science	1995	\$4,000
Asynchronous groupware toolkits	Computer Science	1994	\$2,000

Academic Service

EiC Cttee.	2024	Search committee for ACM ToCHI Editor in Chief.
Grant reviewer	2024	French National Research Agency (ANR).
Grant reviewer	2024	Israel Science Foundation (ISF).
External assessor	2019	Promotion to Full Professor: University of Toronto.
External assessor	2018	Promotion to Full Professor: University of Cambridge.
Rapporteur	2018	Habilitation, Dr Eric Lecolinet, Université Paris Sud, France.
Rapporteur	2018	Habilitation, Dr Marcos Serrano, University Toulouse III Paul Sabbatier, France.
External assessor	2017	Promotion application: University of Waterloo.
Chair	2018–2021	Technical program chair for ACM CHI 2020 Conference
Committee member	2016–2020	Steering Committee for the ACM CHI Conference
Rapporteur	2017	Habilitation, Dr Caroline Appert, University Paris-Sud, France.
Chair	2016–2018	CORE Chris Wallace Research Award Panel
Deputy Chair	2016–2018	PBRF Mathematical and Information Sciences Panel
Editorial	2015–	Editorial Board of Foundations and Trends in Human-Computer Interaction.
Editorial	2015–	Editorial Board of Human-Computer Interaction. Taylor and Francis.
Chair	2012–2015	Papers co-chair for ACM CHI 2014 and 2015 (over 2000 submissions).
Service Award.	2014.	ACM SIGCHI Recognition of Service Award.
Editorial	2012–	Editorial Board of ACM Transactions on Computer-Human Interaction. ACM.
Panel member	2011–2012	Member of the Mathematics, Information Sciences and Technology PBRF panel.
Panel member	2008–2009	BuildIT Postdoctoral Award.
Governance board	2008–2010	Elected member of the governance board of BuildIT.
Panel member	2009, 2011	Member of the Mathematics and Information Sciences Marsden panel.
Panel member	2008–2009	BuildIT Postdoctoral Award.
Panel member	2008–2010	BuildIT Emerging Reseacher Travel Fund.
Mentor	2009–2010	BuildIT mentor for early career researchers.
External assessor	2010	Docent application advisor, University of Tampere, Finland.
External assessor	2010	Promotion application for University of Auckland.
External assessor	2007, 2009	Promotion applications for INRIA, University Paris-Sud.
External assessor	2007	Postdoctoral applicants for Ryerson Univerity, Canada.
Editorial	2002–2013	Associate Editor for International Journal of Human-Computer Studies.
Editorial	2002–2013	Member Editorial Board for Interacting with Computers Journal.
Chair	2010–2011	Subcommittee Chair ACM CHI'11.
Senior Reviewing	2004–	Associate Chair (Papers) for ACM CHI'04, CHI'08, CHI'09.
Senior Reviewing	2009	Associate Chair (Papers) for ACM UIST 2009.
Senior Reviewing	2007	Short papers chair for INTERACT 2007
Chair	2004–2005	Papers chair for Australasian User Interface Conferences
Competition judge	2009	Student Research Competition at CHI 2009.
Journal Reviewing	1993–	Communications of the ACM, Human-Computer Interaction Journal, IEEE Transactions on Visualization and Computer Graphics, ACM Transactions on Computer-Human Interaction, Ergonomics, Information and Software Technology, Information Technology and Society, Interacting with Computers, Virtual Reality, Personal and Ubiquitous Computing, Journal of Computer-Mediated Communication, and International Journal of Human-Computer Studies.
Misc.	2003	Organised the 'Masterclass and Symposium on Interacting with and Through Next Generation Computer Systems', held at the University of Canterbury.
Organising committees	1996, 1998	OzCHI
Doctoral consortium	2008	CHI 2008
Other reviewing	Ongoing	Graphics Interface 2009, 2010, 2011; UIST Demos 2009; ACM CHI 2007, 2006, 2005, 2004, 2003, 2002. AICCS-07; ACM UIST 2007, 2006, 2005, 2003; InfoVis 2006; BEyond time and errors: novel evaLuation methods for Information Visualization (BELIV'06); ACM Advanced Visual Interfaces (AVI) 2006; IEEE Tencon'05; CHINZ'05; Interact'05; MobileHCI'05; ACM CSCW 2004; ACM UIST 2003; INTERACT 2003; Australasian User Interface Conference, 2000–2003; The Active Web, '99; ACM Conference on Data Communications 1994; ACM Conference on Human Factors in Computing Systems 1996; OZCHI'94-97; APCHI'96 & '98; ACM Hypertext '97; Graphics Interface '97; InterAct '97; ACM Group '97.

Invited seminar presentations outside the University of Canterbury

Keynote address, ACM Interactive Surfaces and Spaces Conference, Wellington, NZ.	November, 2022.
Keynote address, INTERACT Conference, Mumbai.	September, 2017.
Dean's Colloquium, ANU, Australia.	August, 2016.
Engineering & Computer Science, ANU, Australia.	August, 2016.
Institute of Advanced Studies, University of Bristol, UK.	August, 2013.
Department of Computer Science, University of Glasgow, UK.	August, 2013.
Colloquium Polaris, INRIA, Lille, France.	July, 2013.
Department of Computer Science, University of Manitoba, Canada.	August, 2010.
Department of Computer Science, University of Manitoba, Canada.	August, 2010.
Department of Computer Science, University of British Columbia, Canada.	August, 2010.
Keynote address, Australasian Computer Science Week.	January, 2010.
Department of Computer Science, University of British Columbia, Canada.	July, 2009.
Department of Computer Science, University of Calgary, Canada.	July, 2008.
Department of Computer Science, Swansea University, UK.	September, 2008.
INRIA, Universite Paris-Sud, France.	September, 2006.
Department of Computer Science, Swansea University, UK.	August, 2006.
Logitech Inc., California.	July, 2006.
Department of Computer Science, Dalhousie University, Nova Scotia.	July, 2006.
Digital Graphics Project Lab, University of Toronto.	June, 2006.
Department of Computer Science, University of Saskatoon.	June, 2006.
Department of Computer Science, University of Calgary.	April, 2006.
IBM Almaden Research Centre.	March, 2006.
Digital Fusion Inc..	February, 2006.
Usability Professionals Association (UPA), Canterbury Innovation Incubator.	November, 2005.
Usability Professionals Association (UPA), Canterbury Innovation Incubator.	November, 2004.
Department of Computer Science, University of Calgary, Canada.	November, 2003.
Human-Computer Interaction Lab, Univ. of Maryland, College Park, USA.	October 2003.
Department of Computer Science, University of Glasgow.	September, 2003.
Department of Psychology, University of York.	September, 2003.
Interaction Centre, University College London.	September, 2003.
Department of Computer Science, University of Southampton.	September, 2003.
New Zealand Computer Society, Christchurch.	June, 2003.
Department of Computer Science, University of Waikato.	December, 2002.
Allied Telesyn Research.	November, 2002.
Department of Computer Science, University of Calgary, Canada.	October, 2001.
Institute for Info. Tech., National Research Center, Ottawa, Canada.	October, 2001.
Grouplab, University of Calgary, Canada.	October, 2001.
Department of Computer Science, University of Saskatchewan, Canada.	October, 2001.
Department of Computer Science, University of Cape Town, South Africa.	September, 2001.
Department of Computer Science, Victoria University of Wellington.	July, 2000.
Inaugural ACM SIGCHI.NZ meeting. Massey University.	June 2000.
Department of Computer Science, University of Waikato.	May, 2000.
Department of Computer Science, University of Calgary.	March, 1999.
Center for LifeLong Learning and Design, University of Colorado, Boulder.	November, 1998.
Norsk Regencentral (Norwegian Research Centre), Oslo, Norway.	August, 1998.
Department of Computer Science, University of Calgary.	January, 1996.
Department of Computer Science, Massey University.	January, 1993.
Department of Computer Science, University of Waikato.	January, 1993.

Visiting academics

I have attracted many world leaders in Human-Computer Interaction to the department. Most of them have presented several seminars while at Canterbury, including numerous ‘Science Prestige Lectures’ and Ben Sheiderman’s Royal Society Evening Lecture.

Prof. Joanna McGrenere	University of British Columbia, Vancouver, Canada.	One month, 2023.
Prof. Carl Gutwin	University of Saskatchewan, Saskatoon, Canada.	Two months, 2023.
Dr. Philip Quinn	Google Research, Mountain View, California.	One year, 2020-2021.
Prof. Carl Gutwin	University of Saskatchewan, Saskatoon, Canada.	Four weeks, 2020.
Prof. Carl Gutwin	University of Saskatchewan, Saskatoon, Canada.	Seven weeks, 2015-2016.
Prof. Wolfgang Stuerzlinger	University of Waterloo, Canada	Three months, 2011-2012
Prof. Kari-Jouko Raiha	University of Tampere, Finland	Nine months, 2009-2010
Dr. Saila Ovaska	University of Tampere, Finland	Nine months, 2009-2010
Dr. David Ahlstrom	Klagenfurt University, Austria.	Eighteen months, 2008-2010.
Prof. Carl Gutwin	University of Saskatchewan, Saskatoon, Canada.	Three months, 2008.
Prof. Karon MacLean	University of British Columbia, Canada.	Six months, 2007-2008.
Prof. Bob Kraut	Carnegie Mellon University.	Erskine Visitor. Three months, 2008.
Prof. Andrew Monk	University of York, UK.	Three months, 2008.
Dr. Gonzalo Ramos	University of Toronto, Canada.	One month, 2007.
Prof. Carl Gutwin	University of Saskatchewan, Saskatoon, Canada.	Erskine Fellow. Four months, 2007.
Prof. Alistair Sutcliffe	University of Manchester, UK.	One month, 2007.
Prof. Carl Gutwin	University of Saskatchewan, Saskatoon, Canada.	One week, 2005.
Prof. Ben Bederson	University of Maryland, Maryland, USA.	One week, 2004.
Prof. Ben Shneiderman	University of Maryland, Maryland, USA.	Erskine Fellow. One month, 2003.
Prof. Jenny Preece	University of Maryland, Baltimore County, Maryland.	Erskine Fellow. One month, 2003.
Prof. Stephen Brewster	University of Glasgow, UK.	Erskine Fellow. Six months, 2003.
Prof. Gerhard Fischer	University of Colorado, Boulder.	One year, 2002–2003.
Prof. Bob Spence	Imperial College London.	Erskine Fellow. One month, 2002.
Prof. Gerhard Fischer	University of Colorado, Boulder.	Erskine Fellow. One month, 2000.
Prof. Harold Thimbleby	Middlesex University, UK.	Erskine Fellow. One month, 1997.
Dr Allen Cypher	Apple Research, Cupertino, California.	One week, 1996.
Dr Akihiko Machizawa	Communications Research Laboratory, Japan.	Three years, 1996–1998.
Professor Brian Wyvill	University of Calgary, Canada.	Erskine Fellow. Two months, 1996.
Professor Saul Greenberg	University of Calgary, Canada.	Erskine Fellow. Six months, 1995.
Carl Gutwin	University of Calgary, Canada.	Two months, 1995.
Dr Steve Jones	University of Abertay, Dundee.	Six months, 1995.

Expert Witness Consulting

Role	Expert witness for SBM Legal, counsel for New Zealand Te Whatu Ora Waitemata. (analysis, reports).
Case	Vinod Chand v Health New Zealand Te Whatu Ora Waitemata. <i>Employment relations authority (regarding text auto-correct functionality).</i>
Dates	May 2024 – June 2024.
Role	Expert witness for Fish & Richardson, counsel for Apple. (analysis, reports).
Case	Smith Interface Technologies v. Apple. <i>Patent Trial and Appeal Board.</i>
Dates	December 2023 –
Role	Expert witness for Finnegan, Henderson, Farabow, Garrett & Dunner, counsel for Samsung.
Role	Expert witness for Fish & Richardson, counsel for Apple. (analysis, reports, deposition).
Case	Masimo v. Apple. <i>Patent Trial and Appeal Board. IPR2023-00664, IPR2024-00290</i>
Dates	July 2023 –
Role	Expert witness for Finnegan, Henderson, Farabow, Garrett & Dunner, counsel for Samsung. (analysis).
Case	Smith Interface Technologies v. Samsung. <i>Eastern District of Texas. 2:22-cv-00290-JRG</i>
Dates	November 2022 – June 2023.
Role	Expert witness for Morrison Foerster, counsel for Apple. (analysis, reports, deposition).
Case	Firstface Co. Ltd. v. Apple Inc. <i>Northern District of California. 3:18-cv-02245-JD.</i>
Dates	June 2022 – March 2024.
Role	Expert witness for Finnegan, Henderson, Farabow, Garrett & Dunner, counsel for Samsung. (analysis).
Case	Village Green Technologies v. Samsung. <i>Eastern District of Texas. 2:21-cv-00323-JRG-RSP</i>
Dates	January 2022 – March 2022.
Role	Expert witness for Desmarais, counsel for IBM. (analysis, reports, deposition).
Case	Chewy, Inc. v. IBM. <i>Southern District of New York. 1:21-cv-01319-JSR.</i>
Dates	August 2021 – Jan 2023.
Role	Expert witness for Farella, Braun & Martel, counsel for Google. (analysis, reports, testimony at trial).
Case	Profectus Technology versus Google. <i>Western District of Texas. 6:20-cv-101-ADA.</i>
Dates	March 2021 – October 2021.
Role	Expert witness for Latham & Watkins and Allen & Overy, counsel for Facebook/Instagram/WhatsApp/Meta. (analysis, reports, deposition).
Case	Facebook/Meta Platforms <i>et al.</i> versus Wrinkl <i>Patent Trial and Appeal Board. IPR2021-00882, -00883</i>
Dates	February 2021 – November 2022.

Role	Expert witness for Morrison & Foerster, counsel for Renesas. (analysis).
Case	Neodron versus Renesas Electronics Corp. <i>Western District of Texas</i> . 6:20-CV-00529.
Dates	December 2020 – August 2021.
Role	Expert witness for DLA Piper, counsel for Samsung. (analysis, declaration).
Case	Neonode versus Samsung. <i>Western District of Texas</i> . 6:20-CV-00507.
Dates	September 2020 – July 2023.
Role	Expert witness for DLA Piper (counsel for Apple, Microsoft and ASUS), Morrison & Foerster (counsel for Amazon), Finnegan, Henderson, Farabow, Garrett & Dunner (counsel for LG), O'Melveny & Myers (counsel for Samsung), Vasquez Benisek & Lindgren (counsel for ASUS). (analysis).
Case	Neodron versus Apple, Microsoft, Amazon & others. <i>US International Trade Commission</i> . 337-TA-1193.
Dates	May 2020 – November 2020.
Role	Expert witness for Desmarais, counsel for IBM. (analysis, reports).
Case	IBM versus Zillow Group. <i>Central District of California</i> . 8:19-CV-01777.
Dates	October 2019 – May 2020.
Role	Expert witness for DLA Piper (counsel for Microsoft, HP, Lenovo), Morrison & Foerster (counsel for Amazon), Finnegan, Henderson, Farabow, Garrett & Dunner (counsel for Motorola and Lenovo), Alston & Bird (counsel for Dell), O'Melveny & Myers (counsel for Samsung). (analysis, reports, deposition).
Case	Neodron versus Microsoft, HP, Amazon, Samsung, Dell, Motorola, Lenovo. <i>US International Trade Commission</i> . 337-TA-1162.
Dates	August 2019 – December 2020.
Role	Expert witness for Jones Day, counsel for Google and Huawei. (analysis).
Case	American Patents versus Huawei.
Dates	April 2019 – July 2019
Role	Expert witness for Gilbert+Tobin, counsel for Aristocrat Technologies. (reports, testimony at trial).
Case	Aristocrat Technologies versus Commissioner of Patents. <i>Federal Court of Australia</i> . NSD 1343/2018.
Dates	September 2018 – June 2020
Role	Expert witness for Williams & Connolly and Alston & Bird, counsel for Google and ASUS Inc. (reports, deposition).
Case	Koninklijke Philips versus ASUS. <i>Northern District of California</i> 4:18-cv-01885-HSG.
Dates	March 2016 – March 2020
Role	Expert witness for Williams and Connolly, counsel for Google Inc. (reports, deposition).
Case	Google versus Philips. <i>Patent Trial and Appeal Board</i> . IPR2017-00386/387/388/389.
Dates	March 2016 – March 2020

Role	Expert witness for DLA Piper, counsel for Apple Inc. (reports, deposition, testimony at trial).
Case	Immersion versus Apple. <i>US International Trade Commission</i> . 337-TA-1004 and 337-TA-990 (consolidated).
Dates	June 2016 – May 2017
Role	Expert witness for DLA Piper, counsel for Apple Inc. (reports).
Case	Apple versus Immersion. <i>Patent Trial and Appeal Board</i> . IPR2016-01777.
Dates	June 2016 – May 2017
Role	Expert witness for Wilmer Cutler Pickering Hale and Dorr, counsel for Apple Inc. (analysis).
Case	Core Wireless Licensing S.a.r.l. v. Apple Inc. <i>Eastern District of Texas</i> . 6:14-cv-00751 and 6:14-cv-00752.
Dates	August 2015 – June 2016
Role	Expert witness for Weil, Gotshal & Manges, counsel for Apple Inc. (analysis).
Case	Motorola Mobility Inc. versus Apple Inc. <i>Southern District of Florida</i> . 10-CV-23580-RNS-TEB and 12-CV-20271-RNS-TEB.
Dates	October 2013 – March 2014
Role	Expert witness for Gibson, Dunn & Crutcher, counsel for Apple Inc. (reports, deposition, testimony at trial).
Case	Apple Inc. versus Samsung Electronics Co. Ltd. <i>Northern District of California</i> . 12-CV-0630-LHK.
Dates	June 2013 – June 2014
Role	Expert witness for Herbert Smith Freehills, counsel for Apple Inc. (reports, testimony at trial).
Case	Apple Inc. versus Samsung Electronics Co. Ltd. <i>Federal Court of Australia</i> . NSD 1243 of 2011.
Dates	September 2011 – May 2013

Other

- Member of the Association of Computing Machinery (ACM).
- Member of the ACM Special Interest Group of Computer-Human Interaction.