

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

LG ELECTRONICS INC.,

Petitioner,

v.

PANTECH CORPORATION

Patent Owner.

Case IPR2023-01268

U.S. Patent No. 9,575,631

**DECLARATION OF VIJAY K. MADISETTI IN SUPPORT OF PETITION
FOR *INTER PARTES* REVIEW OF U.S. PATENT NO. 9,575,631**

TABLE OF CONTENTS

	Page
I. INTRODUCTION.....	1
II. QUALIFICATIONS.....	5
III. LEGAL UNDERSTANDING.....	10
A. My Understanding of Claim Construction.....	10
B. My Understanding of Anticipation.....	12
C. My Understanding of Obviousness.....	13
D. My Understanding of Written Description and Enablement.....	18
E. My Understanding of the Priority Date of a Patent for Prior Art Purposes.....	20
IV. BACKGROUND OF THE TECHNOLOGY	20
V. THE '631 Patent.....	23
A. '631 Patent Overview	23
B. Prosecution History of the '631 Patent.....	25
VI. THE '631 PATENT PRIORITY DATE	27
VII. LEVEL OF ORDINARY SKILL IN THE ART	28
VIII. CLAIM CONSTRUCTION	32
IX. GROUNDS OF UNPATENTABILITY	34
A. Ground 1: Paasonen Anticipates and Renders Obvious Claims 1-10.....	36
1. Overview of Paasonen.....	36
2. Invalidity of Claims 1-10 Over Paasonen	41
(a) Element [1.pre].....	41
(b) Element [1.a].....	45
(c) Element [1.b].....	47

(d)	Element [1.c]	49
(e)	Element [1.d]	55
(f)	Element [1.e]	57
(g)	Element [1.f]	59
(h)	Element [1.g]	67
(i)	Claim [2]	70
(j)	Claim [3]	73
(k)	Claim 4	75
(l)	Claim [5]	77
(m)	Claim [6]	79
(n)	Claim [7]	83
(o)	Claim [8]	86
(p)	Claim [9]	89
(q)	Claim [10]	93
B.	Ground 2: Paasonen in view of Nakano Renders Obvious	
	Claims 1-10	96
1.	Overview of Nakano and Motivation to Modify Paasonen with Nakano's Teachings	96
2.	Invalidity of Claims 1-10 Over Paasonen in view of Nakano	101
(a)	Element [1.c]	101
(b)	Claim [8]	104
(c)	Claim [9]	106
C.	Grounds 3–4: Paasonen in view of Aaltonen (Ground 3) and further in view of Nakano (Ground 4) Render	
	Obvious Claims 4 and 7	109
1.	Overview of Aaltonen and Motivation to Modify Paasonen and Nakano with Aaltonen's Teachings	109
2.	Invalidity of Claims 4 and 7 over Paasonen in view of Aaltonen (Ground 3) and further in view of Nakano (Ground 4)	112
(a)	Claim [4]	112
(b)	Claim [7]	116
D.	Ground 5: Hotelling Anticipates and Renders Obvious	
	Claims 1-4, and 7	118
1.	Overview of Hotelling	118

2.	Hotelling’s Effective Priority Date and Incorporation of Hotelling-521	121
3.	Invalidity of Claims 1–4 and 7 over Hotelling.....	131
(a)	Element [1.pre]	132
(b)	Element [1.a]	135
(c)	Element [1.b]	137
(d)	Element [1.c]	138
(e)	Element [1.d]	141
(f)	Element [1.e]	142
(g)	Element [1.f].....	143
(h)	Element [1.g]	151
(i)	Claim [2].....	155
(j)	Claim [3].....	158
(k)	Claim [4].....	164
(l)	Claim [7].....	165
E.	Ground 6: Hotelling in view of Hellyar Renders Obvious Claims 1–4, 6–10	167
1.	Overview of Hellyar and Motivation to Modify Hotelling with Hellyar’s Teachings.....	167
2.	Invalidity of Claims 1–4, 6–20 over Hotelling in view of Hellyar	173
(a)	Element [1.a]	173
(b)	Element [1.b]	176
(c)	Element [1.c]	178
(d)	Element [1.d]	181
(e)	Claim [6].....	184
(f)	Claim [8].....	187
(g)	Claim [9].....	190
(h)	Claim [10].....	193
F.	Ground 7: Hotelling in view of Hellyar and Tu Renders Obvious Claims 5–6.....	195
1.	Overview of Tu and Motivation to Modify Hotelling in view of Hellyar with Tu’s Teachings.....	195
2.	Invalidity of Claims 5–6 Over Hotelling in view of Hellyar and Tu	199
(a)	Claim [5].....	199

(b)	Claim [6].....	203
X.	SECONDARY CONSIDERATIONS.....	206
XI.	CONCLUSION	206

I. INTRODUCTION

1. I have been retained by LG Electronics Inc. (whom I will refer to as “LGE” or “Petitioner”) to provide my opinions on certain issues related to U.S. Patent No. 9,575,631 (the “’631 Patent”) which I understand has been designated as Exhibit 1001 in connection with the above-captioned *inter partes* review (IPR) proceeding. In particular, I have been asked to provide my insights, analysis, and opinions regarding whether claims 1-10 of the ’631 Patent (“Challenged Claims”) are anticipated by and/or obvious over the prior art references identified below. A listing of the Challenged Claims, including the element numbers assigned to each element in those claims, is attached to this declaration as **Appendix C**.

2. I understand the ’631 Patent is titled “Method for Selecting and Controlling Second Work Process During First Work Process in Multitasking Mobile Terminal,” identifies as its named inventors Seul Ki Choi and Sang Jin Yoon, and is currently assigned to Pantech Corporation (“Patent Owner”). I have considered the ’631 Patent while arriving at my opinions in this declaration.

3. I understand that the file history of the ’631 Patent has been designated as Exhibit 1002. I have considered this file history, and I will refer to it as the “’631 File History” or by its exhibit number.

4. I understand that the ’631 Patent was filed as U.S. Patent Application 15/010,629 on January 29, 2016.

5. I understand that the '631 Patent claims priority to two Korean patent applications: KR 10-2005-0073877 filed August 11, 2005; and KR 10-2005-0102198 filed October 28, 2005.

6. I have considered the prior art cited in my declaration, including:

- **U.S. Patent No 9,785,329** to Paasonen et al. (“Paasonen”) titled “Pocket Computer and Associated Methods,” filed December May 23, 2005 and published October 10, 2017. I understand that a copy of this patent has been designated as Exhibit 1005.
- **U.S. Patent Application Publication No. 2004/0100479** to Nakano et al (“Nakano”) titled “Portable Information Terminal, Display Control Device, Display Control Method, and Computer Readable Program Therefor,” filed May 9, 2003 and published May 27, 2004. I understand that a copy of this patent publication has been designated as Exhibit 1006.
- **U.S. Patent Application Publication No. 2006/0020904** to Aaltonen et al (“Aaltonen”) titled “Stripe User Interface,” filed July 11, 2005 and published January 26, 2006. I understand that a copy of this patent publication has been designated as Exhibit 1007.
- **U.S. Patent Application Publication No. 2005/0257169** to Tu (“Tu”) titled “Control of Background Media When Foreground Graphical

User Interface Is Invoked,” filed June 22, 2004 and published November 17, 2005. I understand that a copy of this patent publication has been designated as Exhibit 1008.

- **U.S. Patent Application Publication No. 2006/0197753** to Hotelling (“Hotelling”) titled “Multi-Functional Hand-Held Device,” filed March 3, 2006 and published September 7, 2006. I understand that a copy of this patent publication has been designated as Exhibit 1009.
- **U.S. Provisional Patent Application No. 60/658777** (“Hotelling-Prov.”) filed March 5, 2005. I understand that Hotelling-Prov. is incorporated by reference into Hotelling and is therefore part of the Hotelling disclosure. I understand that a copy of this provisional patent publication has been designated as Exhibit 1010.
- **U.S. Patent Application Publication No. 2006/0026521** to Hotelling et al. (“Hotelling-521”) titled “Gestures for Touch Sensitive Input Devices,” filed July 30, 2004 and published February 2, 2006. I understand that Hotelling-521 is incorporated by reference into both Hotelling and Hotelling-Prov., and is therefore part of both the Hotelling and Hotelling-Prov. disclosures. I understand that a copy of this patent publication has been designated as Exhibit 1011.

- **U.S. Patent Application Publication No. US2003/0117440** to Hellyar et al. (“Hellyar”) titled “Method And System For Switching Between Multiple Computer Applications,” filed December 21, 2001 and published June 26, 2003. I understand that a copy of this patent publication has been designated as Exhibit 1012.

7. I am being compensated by LGE at my standard hourly consulting rate of \$600 for my time on this matter. My compensation is not dependent on the outcome of this proceeding.

8. In forming my opinions, I relied on the documents cited in this declaration and the documents identified in the attached **Appendix B**. These documents comprise patents, file histories, printed publications, and other related documents. As discussed below, each document is a type that experts in my field would have reasonably relied upon when forming their opinions. Further, experts in my field would have had access to each document either through the applicable patent offices and/or well-known libraries, conferences, or publications in the field. My opinions are also based upon my personal and professional experience.

9. In parallel with this declaration, I have prepared a declaration for IPR2023-01271. I also understand that declarations are being prepared by other experts in IPR2023-01267, IPR2023-01269, IPR2023-01270, IPR2023-01272, and IPR2023-01273. Due to overlap in prior art and subject matter and similarities in

corresponding analyses, I understand that some parts of my declaration may bear superficial similarities to documents filed in these other IPRs, but all analysis and opinions presented herein are my own.

II. QUALIFICATIONS

10. I received my Bachelor of Technology (Honors) in Electronics and Electrical Communication Engineering at the Indian Institute of Technology (IIT) in Kharagpur, India in 1984. I obtained my Ph.D. in Electrical Engineering and Computer Science at the University of California, Berkeley, in 1989. I received the Demetri Angelakos Outstanding Graduate Student Award from the University of California, Berkeley and the IEEE/ACM Ira M. Kay Memorial Paper Prize in 1989.

11. I am a tenured Full Professor in the Colleges of Computing and Engineering at Georgia Tech. I am knowledgeable and familiar with topics including software engineering, web applications, cloud computing, user interface design, data analytics, wireless communications, messaging protocols, microprocessor architecture, hardware, RF, cellular networks, computer engineering, digital signal processing, and associated software and firmware design. I have created and taught undergraduate and graduate courses in hardware and software design for computer science and engineering applications at Georgia Tech for the past thirty years. I have been the principal investigator (“PI”) or co-PI in several active research programs in these areas, including DARPA’s Rapid

Prototyping of Application Specific Signal Processors, the State of Georgia's Yamacraw Initiative, the United States Army's Federated Sensors Laboratory Program, and the United States Air Force Electronics Parts Obsolescence Initiative. I have received an IBM Faculty Award and NSF's Research Initiation Award.

12. I have designed several specialized computer and communication systems over the past two decades at Georgia Tech for tasks such as wireless audio/video processing and messaging/protocol processing for portable platforms, such as cell phones and Personal Digital Assistants. I also developed UIs for tools for electronic design automation of complex chipsets and their reverse engineering processes for U.S. Air Force and Lockheed Martin and Boeing in the late 1990s and early 2000s. I have worked on designing systems that are efficient from performance, size, weight, area, and thermal considerations. I have developed courses and classes for the industry on these topics, and many of my lectures in advanced computer system design, developed under the sponsorship of the United States Department of Defense in the late 1990s, and have been used by several U.S. and international universities as part of their course work. I graduated more than 20 Ph.D. students that now work as professors or in technical positions around the world.

13. Over the years, I have been an active consultant to industry and various research laboratories (including Massachusetts Institute of Technology Lincoln Labs and Johns Hopkins University Applied Physics Laboratory). I have founded three companies in the areas of embedded software, military chipsets involving imaging technology, and wireless communications. The first of these companies, VP Technologies, offers computing systems for helicopter imaging systems for the United States Air Force. The second of these companies, Soft Networks, LLC, offers software for multimedia and wireless computing platforms, including the development of a set-top box for Intel that decodes MPEG-2 video streams and imaging codes for multimedia phones. The third of these companies, Elastic Video, uses region-of-interest based video encoding to capture high quality video at low bit rates, with primary application for wireless video systems. I also have extensive experience developing user interfaces for products. Recent examples of such work include my work developing a crypto exchange prototype, and my work on a web-based document management environment, called CatchUp, which is now commercialized as CatchUp Cloud (<https://catchup.cloud>).

14. I have authored more than sixty refereed journal publications and around forty peer-reviewed conference publications. I have served as associate editor or on the editorial board for technical journals, including IEEE Transactions on Circuits & Systems II, International Journal in Computer Simulation, and

International Journal in VLSI Signal Processing. I have also authored or co-authored several books, including VLSI Digital Signal Processors (IEEE Press 1995) and the Digital Signal Processing Handbook (CRC Press, 1998, 2010).

15. My experience is relevant to this case. I have worked in the area of user interfaces over the past twenty years as part of my extensive development in mobile and portable applications for mobile phones and portable devices since the early 2000 timeframe. I developed commercial products for leading companies, and was part of the team that developed the user interfaces for a Soft Switch, called SNX 8500, that provides voice and VOIP services for enterprises. In addition to authoring textbooks and research papers in the area of wireless messaging interfaces, I have developed web-based document management systems and web-based fintech applications. I have submitted approximately 100 invention disclosures and provisional patents over the past ten years. I am listed as a first inventor on over thirty issued US patents, and am listed as an inventor on several patents related to wireless messaging. Many of my patents also disclose techniques for efficient user interface design including use of multilevel hierarchies and other UI innovations in the area of user interfaces for work environments.

16. I have been extensively involved in the activities of one of the premier standard setting organizations in the world, the IEEE, since the 1980s, and was a member of the IEEE Press Board from 1995-1997. I was the Technical Program

Chair of IEEE ICASSP in 1996. I was also the Associate Editor of IEEE Transactions on Circuits & Systems II from 1994 to 1997. I am also the director of the IEEE Atlanta Chapter, and the author of several drafts of IETF proposals in the area of VOIP and Voice/Video streaming over the internet. I have been elected a Fellow of the IEEE. The Fellow is the highest grade of membership of the IEEE, a world professional body consisting of over 300,000 electrical and electronics engineers, with only one tenth of one percent (0.1%) of the IEEE membership being elected to the Fellow grade each year. Election to Fellow is based upon votes cast by existing Fellows in IEEE.

17. I have also been awarded the 2006 Frederick Emmons Terman Medal by the American Society of Engineering Education for contributions to Electrical Engineering, including authoring a widely used textbook in the design of VLSI digital signal processors. I was awarded VHDL International Best Ph.D. Dissertation Advisor Award in 1997 and the NSF RI Award in 1990. I was Technical Program Chair for both the IEEE MASCOTS in 1994 and the IEEE Workshop on Parallel and Distributed Simulation in 1990. In 1989, I was recognized with the Ira Kay IEEE/ACM Best Paper Award for Best Paper presented at the IEEE Annual Simulation Symposium.

18. I have attached a more detailed list of my qualifications as
Appendix A.

III. LEGAL UNDERSTANDING

19. I have been asked to consider the validity of claims 1-10 of the '631 Patent, which I refer to as the “Challenged Claims.” I understand that for *inter partes* reviews, invalidity must be shown under a preponderance of the evidence standard. I have concluded that each of Challenged Claims is anticipated and rendered obvious by the prior art based on the references described below and as explained herein.

A. My Understanding of Claim Construction

20. I have been informed that patent claims are construed from the viewpoint of a person of ordinary skill in the art of the patent at the time of the invention. I have been informed that patent claims generally should be interpreted consistent with their plain and ordinary meaning as understood by a person of ordinary skill in the art in the relevant time period (i.e., at the time of the purported invention, or the so called “effective filing date” of the patent application), after reviewing the patent claim language, the specification, and the prosecution history (i.e., the intrinsic record).

21. I have further been informed that a person of ordinary skill in the art must read the claim terms in the context of the claim itself, as well as in the context of the entire patent specification. I understand that in the specification and prosecution history, the patentee may specifically define a claim term in a way that

differs from the plain and ordinary meaning. I understand that the prosecution history of the patent is a record of the proceedings before the U.S. Patent and Trademark Office, and may contain explicit representations or definitions made during prosecution that affect the scope of the patent claims. I understand that an applicant may, during the course of prosecuting the patent application, limit the scope of the claims to overcome prior art or to overcome an examiner's rejection, by clearly and unambiguously arguing to overcome or distinguish a prior art reference, or clearly and unambiguously disavowing claim coverage.

22. In interpreting the meaning of the claim language, I understand that a person of ordinary skill in the art may also consider "extrinsic" evidence, including expert testimony, inventor testimony, dictionaries, technical treatises, other patents, and scholarly publications. I understand this evidence is considered to ensure that a claim is construed in a way that is consistent with the understanding of those of ordinary skill in the art at the time of the claimed invention. For example, this can be useful for a technical term whose meaning may differ from its ordinary English meaning. I understand that extrinsic evidence may not be relied on if it contradicts or varies the meaning of claim language provided by the intrinsic evidence, particularly if the applicant has explicitly defined a term in the intrinsic record.

B. My Understanding of Anticipation

23. I understand that a patent claim is invalid as “anticipated” if all of the limitations of the claim are present, either expressly or inherently, in a single previous device, or described, either expressly or inherently, in a single prior art publication or patent. I understand that, in the relevant time period, 35 U.S.C. § 102 stated that a person shall be entitled to a patent unless, among other bases:

- The invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for patent, or
- The invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of the application for patent in the United States, or
- The invention was described in—(1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for the purposes of

this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

24. I also understand that a prior art reference can disclose a claimed feature because the feature is expressly described or because the feature is inherent in the disclosure. I understand that something is inherent in a prior art reference if the missing descriptive matter must necessarily be present, not merely probably or possibly present, and it would be so recognized by a person of ordinary skill in the art as of the priority date of the patent. I understand that it is acceptable to examine evidence outside of the prior art reference (i.e., extrinsic evidence) in determining whether a feature, while not expressly discussed in the reference, is necessarily present in it.

C. My Understanding of Obviousness

25. I understand that a claim may be invalid if the subject matter described by the claim as a whole would have been obvious to a hypothetical person of ordinary skill in the art in view of a prior art reference or in view of a combination of references at the time the claimed invention was made. Therefore, I understand that obviousness is determined from the perspective of a hypothetical person of ordinary skill in the art and that the asserted claims of the patent should

be read from the point of view of such a person at the time the claimed invention was made. I further understand that a hypothetical person of ordinary skill in the art is assumed to know and to have all relevant prior art in the field of endeavor covered by the patent in suit.

26. I have been informed that there are two criteria for determining whether prior art is analogous and thus can be considered prior art: (1) whether the art is from the same field of endeavor, regardless of the problem addressed, and (2) if the reference is not within the field of the patentee's endeavor, whether the reference still is reasonably pertinent to the particular problem with which the patentee is involved. I have also been informed that the field of endeavor of a patent is not limited to the specific point of novelty, the narrowest possible conception of the field, or the particular focus within a given field. I have also been informed that a reference is reasonably pertinent if, even though it may be in a different field from that of the patentee's endeavor, it is one which, because of the matter with which it deals, logically would have commended itself to a patentee's attention in considering their problem.

27. I have also been advised that an analysis of whether a claimed invention would have been obvious should be considered in light of the scope and content of the prior art, the differences (if any) between the prior art and the

claimed invention, and the level of ordinary skill in the pertinent art involved. I understand as well that a prior art reference should be viewed as a whole.

28. I have also been advised that in considering whether a claimed invention could be obvious over a combination of prior art references, I may assess whether there are apparent reasons to combine known elements in the prior art in the manner claimed in view of interrelated teachings of multiple prior art references, the effects of demands known to the design community or present in the marketplace, and/or the background knowledge possessed by a person of ordinary skill in the art. I understand that other principles may be relied on in evaluating whether a claimed invention would have been obvious, and that these principles include the following:

- A combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results;
- When a device or technology is available in one field of endeavor, design incentives and other market forces can prompt variations of it, either in the same field or in a different one, so that if a person of ordinary skill can implement a predictable variation, the variation is likely obvious;

- If a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious unless its actual application is beyond their skill;
- An explicit or implicit teaching, suggestion, or motivation to combine two prior art references to form the claimed combination may demonstrate obviousness, but proof of obviousness does not depend on or require showing a teaching, suggestion, or motivation to combine;
- Market demand, rather than scientific literature, can drive design trends and may show obviousness;
- In determining whether the subject matter of a patent claim would have been obvious, neither the particular motivation nor the avowed purpose of the named inventor controls whether the claim is obvious;
- One of the ways in which a patent's subject can be proved obvious is by noting that there existed at the time of invention a known problem for which there was an obvious solution encompassed by the patent's claims;

- Any need or problem known in the field of endeavor at the time of invention and addressed by the patent can provide a reason for combining the elements in the manner claimed;
- “Common sense” teaches that familiar items may have obvious uses beyond their primary purposes, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle;
- A person of ordinary skill in the art is also a person of ordinary creativity, and is not an automaton;
- A patent claim can be proved obvious by showing that the claimed combination of elements was “obvious to try,” particularly when there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions such that a person of ordinary skill in the art would have had good reason to pursue the known options within his or her technical grasp; and
- One should be cautious of using hindsight in evaluating whether a claimed invention would have been obvious.

29. I further understand that, in making a determination as to whether the claimed invention would have been obvious to a person of ordinary skill, the Board may consider certain objective factors if they are present, such as: commercial

success of products practicing the claimed invention; long-felt but unsolved need; teaching away; unexpected results; copying; and praise by others in the field. These factors are generally referred to as “secondary considerations” or “objective indicia” of nonobviousness. I understand, however, that for such objective evidence to be relevant to the obviousness of a claim, there must be a causal relationship (called a “nexus”) between the claim and the evidence and that this nexus must be based on a novel element of the claim rather than something in the prior art. I also understand that even when they are present, secondary considerations may be unable to overcome primary evidence of obviousness (such as motivation to combine with predictable results) that is sufficiently strong.

D. My Understanding of Written Description and Enablement

30. I understand that the written description and enablement requirements are set forth in Pre-AIA 35 U.S.C. §112. This section provides:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.

31. I have been informed that to have an adequate written description, the patent must convey with reasonable clarity to a person skilled in the art that, as of the filing date sought, the named inventor was in possession of the invention.

32. I have also been advised that, to demonstrate possession of the invention, the specification must permit a person with ordinary skill in the art to visualize or recognize the identity of the subject matter purportedly described. Although the specification need not describe the claimed subject matter verbatim to demonstrate possession of the invention, generalized language will not suffice if it does not convey the detailed identity of an invention.

33. I have also been advised that, to have an adequate written description, each limitation of the claimed invention must be disclosed expressly or inherently in the specification. It is not sufficient that the missing claim limitation may be obvious from the specification.

34. To be an enabling disclosure, the specification must adequately disclose to a person skilled in the art how to make or carry out the claimed invention without undue experimentation.

35. The omission of minor details from a patent specification's written description does not cause the specification to fail to meet the enablement requirement. However, when there is no disclosure of any specific starting material or of any of the conditions under which a process can be carried out, undue experimentation is required and there is a failure to meet the enablement requirement that cannot be rectified by asserting that all the disclosure related to the process is within the skill of the art.

E. My Understanding of the Priority Date of a Patent for Prior Art Purposes

36. I understand that for a prior art patent or patent application to be entitled to the priority date of a provisional application to which that patent or patent application claims priority, the specification of the provisional must contain a written description of the invention and the manner and process of making and using it, in such full, clear, concise, and exact terms, to enable a person of ordinary skill in the art to practice the invention claimed in the non-provisional application. To satisfy this requirement, I understand that the disclosure of the provisional application must provide enabling written description support for both the claims of the patent and the subject matter of the patent relied upon for prior art purposes.

37. My understanding is that, similarly, a patent can claim priority to an earlier filing date of a foreign patent application if the patent expressly claims the benefit of the earlier filing date, and if disclosure of the foreign patent application provides support for the claims of the patent. I also understand, however, that a patent reference cannot claim the benefit of a filing date of a foreign priority document to establish the patent as prior art.

IV. BACKGROUND OF THE TECHNOLOGY

38. As discussed in Section V, the purported invention of the '631 Patent is methods for “controlling a plurality of work processes in a multitasking mobile

terminal.” ’631 patent, Abstract, 1:23-25. Specifically, the ’631 Patent discloses a method where a user may “select[] a second work process” on a mobile terminal, and the mobile terminal “concurrently performs a first work process and the second work process” without having to terminate the running first work process. ’631 patent, 3:64-67. Specifically, claim 1, which is the only independent claim in the Challenged Claims from which each of the other Challenged Claims depends, recites that the mobile terminal “display[s] one icon in a designated portion of a display . . . correspond[ing] to the first work process” and in response “to a first command for sequentially displaying icons, performing sequential display of icons . . . wherein the icon corresponding to the second work process is displayed.” ’631 patent, Claim 1. It further recites when the icon for the second work process is selected, “replacing the screen of the first work process with a screen of the second work process.” ’631 patent, Claim 1. And it recites that a “third command . . . activate[s] a control menu for controlling . . . function[s] of the first work process.” ’631 patent, Claim 1.

39. Based on my experience in the field, however, the claimed features of the ’631 patent were well-known in the art prior to the ’631 patent’s priority date. One example of such prior art is U.S. Patent No. 9,785,329 (“Paasonen”). I understand that a copy of Paasonen has been designated as Exhibit 1005. Paasonen teaches a mobile terminal with a list of icons that represent the most

recently active (i.e., most recently opened or accessed) “running applications” on a mobile device, when a user selects one of the icons the display of the mobile terminal switches to the selected application. Paasonen, 12:56-13:7, 14:22-37. The list of icons is displayed to the user sequentially and in a “historical order” based on when each application was last active, with the icon for the most-recently active application displayed on top, and the icons below representing the next recently active applications. See Paasonen 8:58-9:4, 9:5-10, 14:60-15:18; Figs. 12-13.

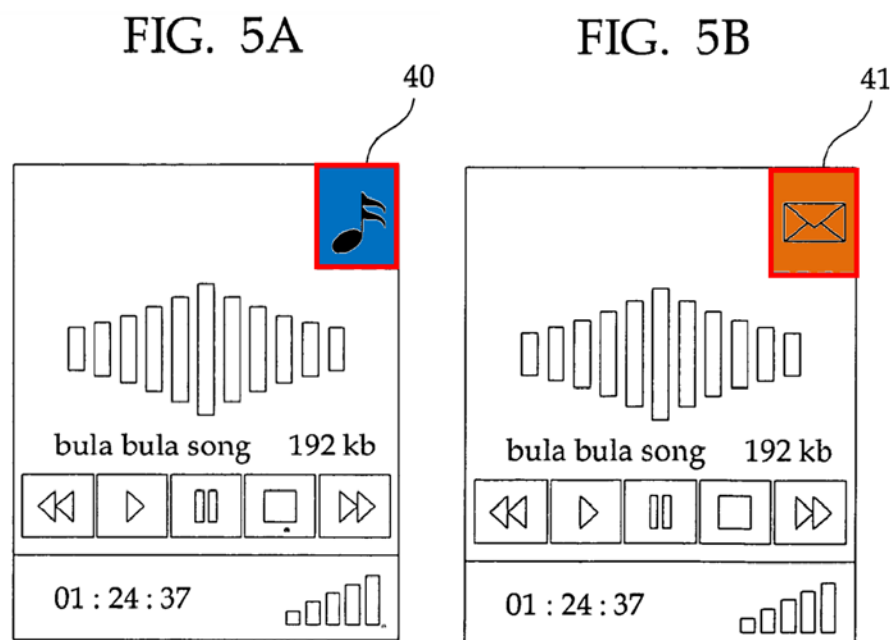
40. Another relevant example of such prior art is U.S. Patent Publication No. 2006/0197753 (“Hotelling”). I understand that a copy of Hotelling has been designated as Exhibit 1009. Hotelling teaches that a hand-held device presents icons to the user that correspond with different programs on the device and when the icons are selected, the device activates the corresponding GUI for the program. See Hotelling-Prov. (Ex. 1010), 18-19:21. To navigate through the options, and select a desired GUI for a program on the device users “flip[]” or “scroll[]” through GUIs presented sequentially. Hotelling-Prov., 18:19-33. U.S. Patent Publication No. 2003/0117440 (“Hellyar”) (Ex. 1012) also presents a method for switching between applications by sequentially “cycling” through icons and preview windows that represent different applications available on a mobile device. Hellyar [0013], [0024], [0045]; Fig. 3C.

41. As I discuss below, the Challenged Claims are anticipated and rendered obvious by the prior art. Further, there were no technological barriers to combining these elements to form the claimed invention and a person of ordinary skill in the art would have been motivated to combine these elements and would have understood that the combination would have yielded predictable results.

V. THE '631 PATENT

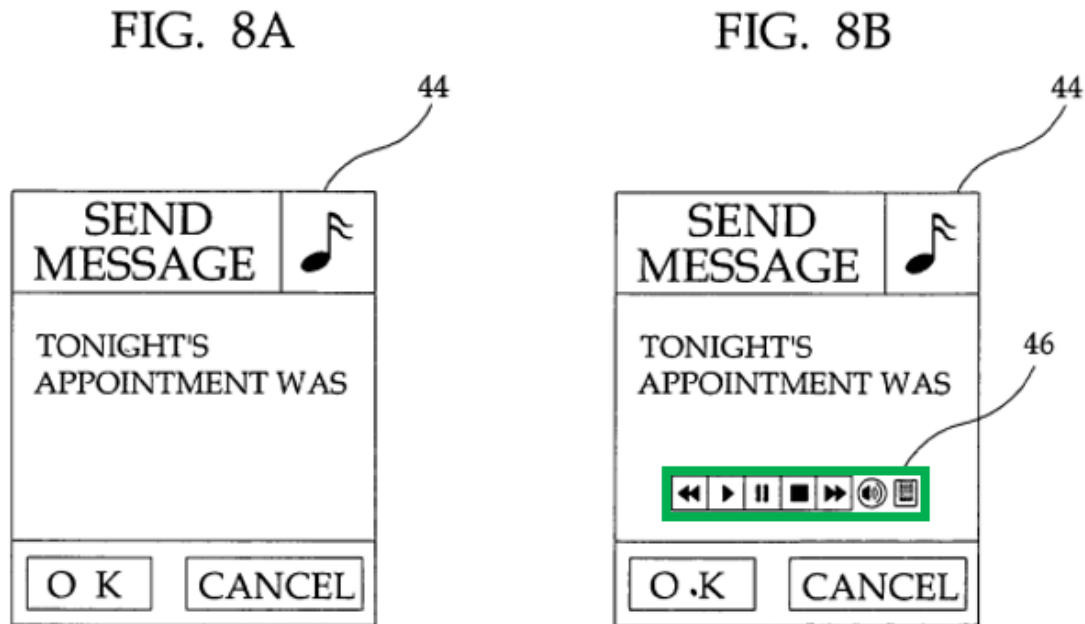
A. '631 Patent Overview

42. The '631 Patent is titled “Method for Selecting and Controlling Second Work Process During First Work Process in Multitasking Mobile Terminal.” Challenged Claims 1-10 of the '631 Patent generally recite a “multitasking mobile terminal” that receives user commands to execute “a predetermined function of a **second** work process during a **first** work process.” '631 patent, Claim 1, 2:59-67, 3:1-25. Figs. 5A-5B show an embodiment of the mobile terminal display in which icons corresponding to a first work process, “**MP3 play**,” and a second work process, “**SMS** message creation,” are displayed in a **designated portion** (“portion 40” and “41” in Figs. 5A and 5B, respectively) of the display:



'631 patent, Figs. 5A-5B, 4:23-24, 4:31-37. The terminal accepts user commands to perform functions including triggering a selected work process, sequentially displaying work process icons in the designated portion, and activating a work process control menu. '631 patent, 2:59-67. For example, selecting the work process icon displayed in the designated portion causes an "activated screen" of the work process to be displayed. '631 patent, 4:1-27, 4:34-45. Selecting the "SMS" icon, for instance, opens the "SMS message creation" screen. '631 patent, 2:63-67, 4:38-45. A user command to "sequentially display[]" icons" causes the **first** and **second work process** icons in the **designated portion** to be displayed "in turn." '631 patent, 5:1-16. Taking the "MP3 play" and "SMS" work processes as examples, the "**MP3 play**" and "**SMS**" icons will be alternately displayed in the **designated portion**. '631 patent, 5:1-16.

43. To control a work process operating in the background while another work process is displayed on the screen, a user may input a command to activate a “**control menu**,” such as the “pop-up menu 46” displayed in Figs. 8A-8B:



'631 patent, 3:7-12, 6:15-25. The control menu in Figure 8B, displayed in response to a user command to activate the menu, “control[s] the function of the MP3 play process” by, for example, controlling music playback functions using play or pause buttons. '631 patent, 5:47-48, 6:21-25.

B. Prosecution History of the '631 Patent

44. The '631 Patent matured from U.S. App. No. 15/010,629, filed on Jan. 29, 2016. The application claims priority from Korean application Nos. 10-2005-

0073877 and 10-2005-0102198, filed respectively on August 11, 2005 and October 28, 2005. '631 File History, 9–17.

45. In an Office Action dated May 19, 2016, the Examiner rejected all pending claims over prior art. '631 File History, 104–26. The Examiner found pending claims 1–4 and 7–9 obvious over Kokubo (U.S. Pat. No. 7,123,945) in view of Cain (U.S. Pub. No. 2005/0183017); pending claims 5–6 obvious over Kokubo and Cain further in view of King (U.S. Pub. No. 2005/0229103); and pending claims 10–19 obvious over Kokubo in view of King. '631 File History, 109–25.¹ After an interview with the Examiner in which the Examiner proposed claim amendments for allowance, the Applicant responded on September 19, 2016 by amending the pending independent claims as shown below:

displaying ~~[[an]]~~one icon ~~corresponding to the first work process~~ in a designated portion of a display, wherein the displayed icon corresponds to the first work process and is displayed together with at least a part of a screen image of the first work process in the display;

if the displayed icon corresponding to the first work process is selected, displaying a screen of the first work process;

in response to a first command for ~~selecting the second work process~~ sequentially displaying icons, performing

¹ In that same office action, all claims were also subject to non-statutory obviousness double patenting rejections over U.S. Pat. Nos. 9,122,518 and 9,250,957, in response to which the Applicant filed a terminal disclaimer. '631 File History, 106–08; '631 File History, 157–61, 171–72.

sequential display, wherein the icon ... is displayed in
[[a]]the designated portion....

'631 File History, 164–74. During an Examiner-initiated interview on September 28, 2016, the Examiner proposed removing the highlighted language above and adding the phrase “in the display” after “displaying a screen of the first work process,” resulting in the above claim elements being amended to reflect the language in claim elements [1.a]–[1.c] of issued claim 1. '631 File History, 183–96. On September 29, 2016, the Examiner entered an Amendment addressing this change after receiving approval from the Applicant. '631 File History, 183–96. The Examiner issued a Notice of Allowance on October 7, 2016 and the '631 Patent issued on February 21, 2017. 188, 234.

VI. THE '631 PATENT PRIORITY DATE

46. I understand the '631 Patent was filed on filed on January 29, 2016. I have not formulated any opinion about the correct priority date for the '631 Patent. For purposes of this declaration, I was asked to assume, with respect to each of the prior art Grounds discussed in this declaration, that the '631 Patent is entitled to the date of the earliest Korean priority application listed on the '631 Patent, namely, August 11, 2005. I have been instructed to consider the patent and the prior art from the perspective of a person of ordinary skill in the art as of August 11, 2005. However, to the extent that the '631 Patent is not entitled to a different priority

date, my opinions herein would remain the same so long as the prior art references that I have identified in this declaration qualify as prior art.

VII. LEVEL OF ORDINARY SKILL IN THE ART

47. In rendering the opinions set forth in this declaration, I was asked to consider the patent claims and the prior art through the eyes of a person of ordinary skill in the art at the time of the alleged invention. I understand that the factors considered in determining the ordinary level of skill in a field of art include the level of education and experience of persons working in the field; the types of problems encountered in the field, the teachings of the prior art, and the sophistication of the technology at the time of the alleged invention. I understand that a person of ordinary skill in the art is not a specific real individual, but rather is a hypothetical individual having the qualities reflected by the factors above. I understand that a person of ordinary skill in the art would also have knowledge from the teachings of the prior art, including the art cited below.

48. With respect to all of the Grounds, I have been asked to assume that the '631 Patent is entitled to the priority date of August 11, 2005. Upon consideration of these factors, in my opinion, on or before this priority date, a person of ordinary skill in the art relating to the technology of the '631 Patent would have had a minimum of a bachelor's degree in electrical or computer engineering or computer science, or a related field, along with at least two years of

professional experience with mobile terminal design. *See, e.g.*, '631 Patent, 1:20-2:28 (summarizing the “Field” and “Background” of the '631 patent, discussing “advance[s] of mobile communication technology” resulting in “various kinds of mobile terminals such as hand-held phones and personal digital assistants (PDAs)” being “commercially available”). Additional graduate education could substitute for professional experience, or significant experience in the field could substitute for formal education.

49. I am qualified to opine concerning what a person of ordinary skill in the art would have known and understood at the time of the '631 Patent's purported invention. Well before August 11, 2005, my level of skill in the art was at least that of a person of ordinary skill in the art. My analysis and conclusions herein are from the perspective of a person of ordinary skill in the art as of August 11, 2005.

50. I understand that in the pending district court litigation, Patent Owner has submitted the Claim Construction Expert Report of Charles Mauro CHFP, who has attested that a person of ordinary skill in the art should be defined as follows:

[A] person of ordinary skill in the art would be an individual with either undergraduate or graduate level education in human factors engineering theory and practice with specific training in human computer interaction design for modern mobile devices including wireless communications terminals. They should have a minimum of one year of industry experience in the

design and user testing of modern mobile GUI systems
and have designed production modern mobile GUI
utilizing common screen design interaction elements.

Mauro Decl. (Ex. 1032) at ¶27. Even under Mr. Mauro's definition, I would have qualified as a person of ordinary skill in the art. I am familiar with user interface design, particularly in the context of mobile devices and communication terminals, and my expertise in developing high-quality hardware and software products over the last thirty-plus years included both graduate education and relevant experience as discussed in Section II. This includes, for example, my work developing interfaces in the late 1990s and early 2000s for professional-quality tools for electronic design automation of complex chipsets and their reverse engineering processes for the U.S. Air Force, Lockheed Martin, and Boeing, as well as my work relating to companies including Soft Networks, LLC and CatchUp, which included experience developing human factors-related aspects of technology for modern mobile devices. Additionally, I developed mobile device user interfaces with Ericsson in the 2002-2006 timeframe, which included developing human factors-related aspects of both mobile device hardware and software applications. I have also considered whether my opinions would differ under Mr. Mauro's proposed definition of a person of ordinary skill in the art, and I have concluded that his proposed definition would not materially affect my opinions set forth in this declaration.

51. While in my opinion any difference between the proposed definitions of a person of ordinary skill in the art is immaterial to my opinions in this declaration, to the extent Patent Owner or Mr. Mauro disputes my definition of a person of ordinary skill, I maintain that my proposed definition accurately defines a person of ordinary skill in the art. The '631 Patent states that the purported invention "relates to a method for controlling a plurality of work processes in a multitasking mobile terminal, and more particularly, to a method for selecting a second work process during a first work process and controlling a predetermined function of the selected second work process." '631 Patent, 1:23-28. The claims of the '631 patent recite features for which a person of ordinary skill in the art would have had a degree in electrical or computer engineering or computer science, or a related field, and are less relevant to other aspects of human computer interaction design. For example, functionalities such as a device performing multiple work processes, and responding to commands to control those work processes as recited in claim 1, as well as specification discussions relating to those elements (including, for example, '631 Patent, 4:12-22, 5:1-35, 5:46-6:6, 6:29-65), describe subject matter that better falls within my definition of a person of ordinary skill in the art, and is less relevant to other aspects of human computer interaction design. Moreover, it would have been well-understood to persons qualifying as persons of ordinary skill in the art under my definition that mobile terminal design includes

human computer interaction design work in order to develop and implement the user interface for such a device.

52. Accordingly, while my definition more accurately defines the relevant person of ordinary skill in the art, the differences between these definitions do not materially impact my opinions set forth in this declaration.

VIII. CLAIM CONSTRUCTION

53. I have been instructed that the Challenged Claims of the '631 Patent are to be given their ordinary and customary meaning as understood by one of ordinary skill in the art at the time of the alleged invention in light of the specification and the prosecution history pertaining to the patent. I have followed these principles in my analysis set forth in this document.

54. I understand that there is a pending district court litigation involving the '631 Patent, and that the parties in that litigation have proposed the following constructions of certain terms in the Challenged Claims:

Term	'631 Patent Limitations	Patent Owner's Construction	Petitioner's Construction
“sequentially displaying icons” / “performing sequential display of icons”	[1.c], [8]	Plain and ordinary meaning	“displaying the next icon in place of the prior icon in a sequence”
“displaying one icon in a designated portion of a display”	[1.a]	Plain and ordinary meaning	“displaying a single icon in a designated portion of a display”

Term	'631 Patent Limitations	Patent Owner's Construction	Petitioner's Construction
"control menu"	[1.f]–[1.g], [3]	Plain and ordinary meaning (e.g., a pop-up menu, a menu window, or the like)	Plain and ordinary meaning
"predetermined function"	[1.pre], [1.f]–[1.g], [2]	Plain and ordinary meaning (i.e., a function among those selected in advance)	Plain and ordinary meaning

Parties' Joint Claim Construction and Prehearing Statement (Ex. 1027), 6–12.

55. I do not express any opinions on the accuracy of these proposed constructions. However, the parties' proposed constructions are not material to my opinions set forth in this declaration because the claims are rendered obvious under any reasonable construction of the claims for the reasons set forth herein. I reserve the right to supplement my opinions in this declaration to the extent that the parties later propose claim constructions for any claim terms in the '631 Patent.

56. I understand the preamble of a claim may or may not be limiting. For purposes of this declaration, I have not been asked to formulate any opinion as to

whether any claim preamble is limiting, and have not formulated any such opinion. Instead, I have been instructed to consider whether the prior art discloses the preambles, to the extent they are limiting. To the extent they are limiting, they are disclosed by the prior art for the reasons that I discuss in Section IX.

IX. GROUNDS OF UNPATENTABILITY

57. As I explain above in Section IV, multitasking mobile terminals with interfaces that permit users to switch between and control work processes without exiting a running work process were well-known in the art.

58. In this section, I will explain why combinations of teachings from the following references render obvious each of the Challenged Claims:

Name	Patent/ Title	Exhibit	Priority	Published/ Issued
Paasonen	US 7,85,329	1005	5/23/2005	10/10/2017
Nakano	US2004/0100479	1006	5/9/2003	5/27/2004
Aaltonen	US2006/0020904	1007	7/11/2005	1/26/2006
Tu	US2005/0257169	1008	6/22/2004	9/7/2006
Hotelling	US2006/0197753	1009	3/4/2005	9/7/2006
Hotelling-521	US2006/0026521	1011	7/30/2004	2/2/2006
Hellyar	US2003/0117440	1012	12/21/2001	6/26/2003

I have been instructed to assume that each of these references qualifies as prior art.

59. As I discuss in greater detail in Section IX.D, Hotelling incorporates App. 60/658777 (“Hotelling-Prov.”) by reference, making Hotelling-Prov part of Hotelling’s disclosure. Hotelling, [0001]. Additionally, both Hotelling and Hotelling-Prov. each incorporate by reference Application No. 10/903,964, published as US2006/0026521 (“Hotelling-521”), making Hotelling-521 part of the disclosure of both Hotelling and Hotelling-Prov. Hotelling [0002]; Hotelling-Prov. 1:8-22. I have been instructed to treat Hotelling-521 as part of Hotelling’s and Hotelling-Prov.’s disclosure. But even if Hotelling-521 were considered a separate reference, a person of ordinary skill in the art would have been motivated to modify Hotelling’s (and Hotelling-Prov.’s) teachings with Hotelling-521’s teachings in light of the explicit incorporation of Hotelling-521 into Hotelling and Hotelling-Prov.; my analysis therefore would not change if Hotelling-521 were treated as a separate reference.

60. As I explain below, the Challenged Claims are invalid based upon the following prior art grounds:

- **Ground 1:** Challenged Claims 1-10 of the ’631 Patent are anticipated or rendered obvious by Paasonen.
- **Ground 2:** Challenged Claims 1-10 of the ’631 Patent are rendered obvious by Paasonen in view of Nakano.

- **Ground 3:** Challenged Claims 4, 7 of the '631 Patent are rendered obvious by Paasonen in view of Aaltonen.
- **Ground 4:** Challenged Claims 4, 7 of the '631 Patent are rendered obvious by Paasonen in view of Aaltonen and Nakano.
- **Ground 5:** Challenged Claims 1–4, 7 of the '631 Patent are anticipated or rendered obvious by Hotelling.
- **Ground 6:** Challenged Claims 1–4, 6–10 of the '631 Patent are rendered obvious by Hotelling in view of Hellyar.
- **Ground 7:** Challenged Claims 5–6 of the '631 Patent are rendered obvious by Hotelling in view of Hellyar and Tu.

61. It is my opinion that Challenged Claims of the '631 Patent are unpatentable as explained below. I note that in this discussion, I may focus on the specific language in the claims under discussion, but it should be understood that my analysis applies to the corresponding elements of all Challenged Claims containing identical or substantially identical language.

A. Ground 1: Paasonen Anticipates and Renders Obvious Claims 1-10

1. Overview of Paasonen

62. Paasonen is titled “Pocket computer and associated methods” and relates to a mobile terminal that includes a “touch-sensitive . . . user interface” through which a user can easily “switch” between any “running applications.”

Paasonen, Abstract, 14:22-29, 15:19-33. A “typical display screen layout” is presented to the user for control over the “running applications,” such as “Internet/WWW/WAP browser,” “contacts,” “email, SMS, MMS,” “calendar,” “organizer,” “video game,” “voice memo,” “alarm clock,” “music player,” “media streaming,” and “control panel” applications. Paasonen, 11:48-65, 14:22-29, 12:44-49. Figure 12 illustrates an embodiment of the screen layout:

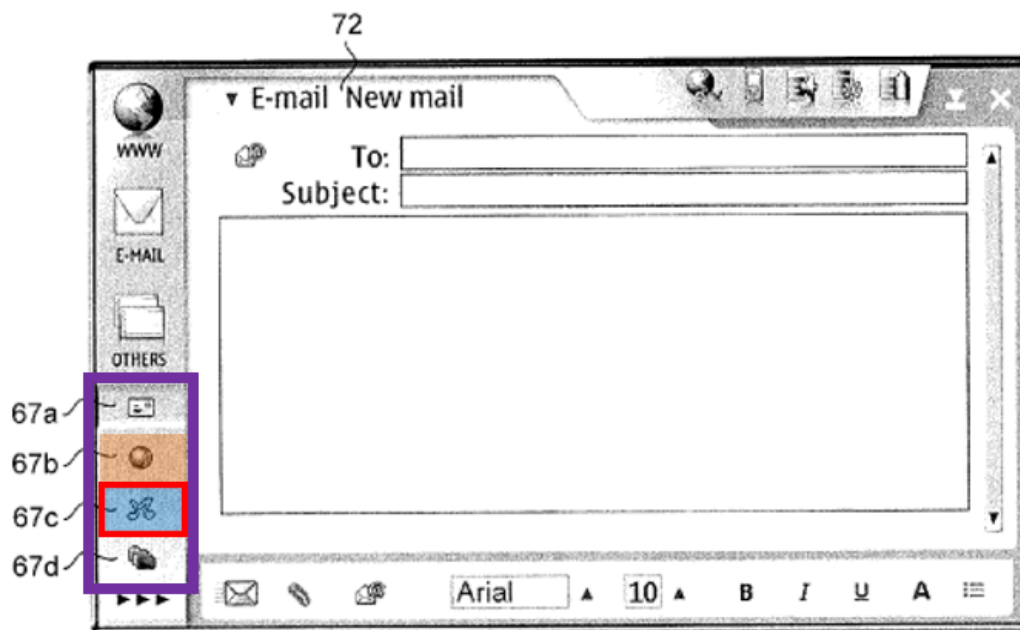


Fig 12

Paasonen, Fig. 12. The display is divided into separate areas including a “task navigator” section, and an “application area” where a “currently active application” displays its content to the user. Paasonen, 12:56-65. The “task navigator” section presents icons to the user that represent “launched applications” and is divided into an “upper portion and a lower portion.” Paasonen, 13:16-52, 14:22-29. The lower

portion of the “task navigator” includes an “**application switcher panel**” that displays icons representing “launched applications” that are currently running, for example the “**browser**” and “**music player**” application icons shown in Figure 12. Paasonen, Figure 12, 11:48-65, 14:22-29. The “**application switcher panel**” presents the icons to the user in a “vertical order” that represents a “historical order” of how recently each application was active. Paasonen, 14:60-15:18, Fig. 12. The most-recently active application’s icon is in the topmost position of the panel and the fourth most-recently active is in the lowest position. Paasonen, 14:60-15:18, Fig. 12. For example, in Figure 12, the **browser** application icon in the second position 67b in the “vertical order” indicates it was “active before” the currently active “messaging application,” whose corresponding icon is shown in the topmost icon position. Paasonen, Fig. 12, 14:60-15:18. The currently active messaging application has its content displayed in the “application area.” Paasonen, 12:56-65. The **music player** application, represented by the music player icon, is the third most recently opened application and is accordingly third in the “vertical order.” Paasonen, 12:56-65. A user issues a command to “switch” the active application by “tapping” an icon on the “application switcher panel.” Paasonen, 14:22-48, 14:60-15:18. Selecting a new application as the active application triggers the remaining icons to be “updated accordingly” with the icon corresponding to the newly selected application moving to the top position in the

switcher panel and the remaining icons sliding down in the “vertical order” accordingly. Paasonen, 14:22-48, 14:60-15:18.

63. Paasonen further teaches the same display includes a “home” screen that displays “downscaled” “**application views**” that “provide access” to “limited” application controls and functions. Paasonen, 15:61-16:40, Fig. 7. A user can “configure which application views to include in the home application, and some particulars of them.” Paasonen, 15:61-16:40. For example, as shown below in Figure 7, the “Internet Radio” limited **application view** provides access to a play function and “Manage” button for controlling the application’s media playback:

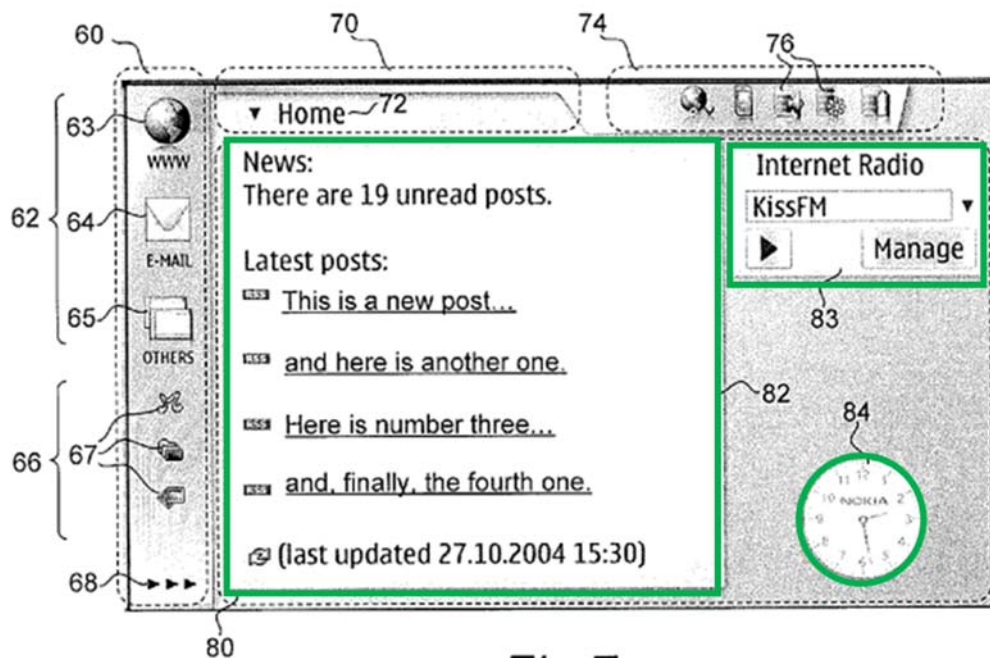


Fig 7

Paasonen, 15:61-16:40, Fig. 7. Thus, Paasonen’s application view not only renders obvious, but affirmatively discloses, a user including home screen views for other

applications, beyond the ones illustrated in Figure 7. Figure 7 further illustrates an “upper portion 62” of the “task navigator” that displays icons 63-65. Paasonen, 13:16-52. These icons permit the user the ability to operate in a “task-oriented manner.” Paasonen, 13:16-52. Selecting one of these icons activates a pop up “**task-oriented menu 90**,” as shown in Figure 8, for different applications:

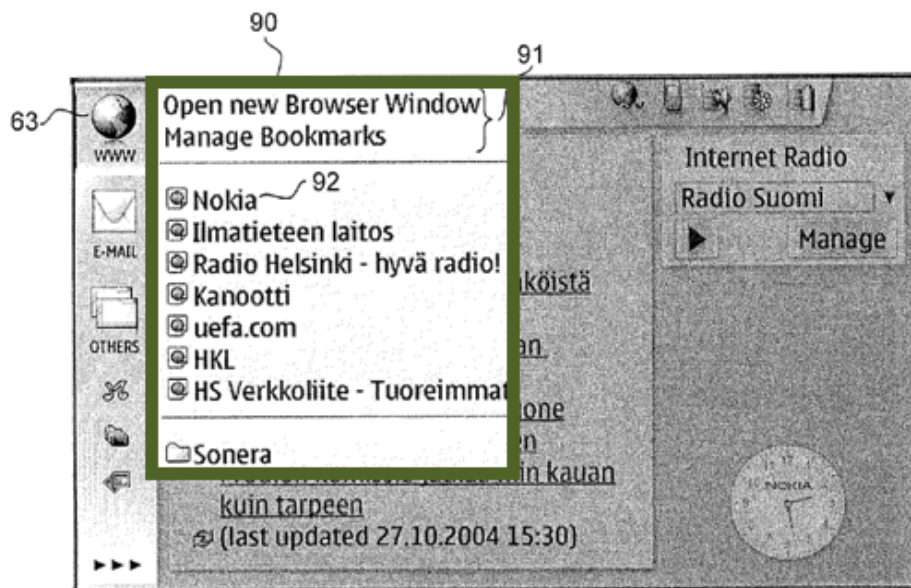


Fig 8

Paasonen, 13:16-52, 14:14-17, Fig. 8. The menu is “context-specific” which means that its menu options relate to the selected parent icon. See Paasonen, 13:16-52. Selecting any function from the “task-oriented menu” will “automatically launch/switch to” the selected application and activate the “appropriate functionality.” Paasonen, 13:53-14:13. Further, as shown in Figures 8 and 10-11, menus—including the pop-up “task-oriented menu”—are displayed

brighter than the application area screen in the background. Paasonen, Fig. 8, 13:4-7, 13:24-14:13. *See also* Paasonen, Figs. 10–11.

2. Invalidity of Claims 1-10 Over Paasonen

(a) Element [1.pre]

64. Element [1.pre] recites “A method for controlling a predetermined function of a second work process during a first work process in a multitasking mobile terminal that performs the first work process and the second work process.” This element is met by Paasonen.

65. Paasonen discloses a method for controlling a predetermined function of a second work process during a first work process (*e.g.*, “access[ing]” a “functionality of a[n]...application,” while running another application) in a multitasking mobile terminal that performs the first work process and the second work process (*e.g.*, “mobile terminal” runs “a currently active application” (*e.g.*, “browser”) and another application (*e.g.*, “audio” or “music player”)). *See, e.g.*, Paasonen (emphasis added):

- Paasonen, Fig. 2

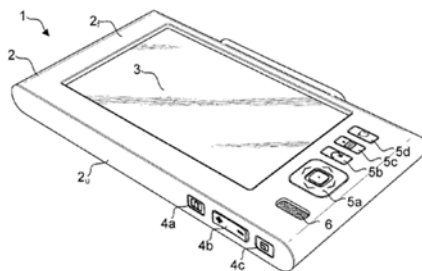


Fig 2

- Paasonen, Fig. 3

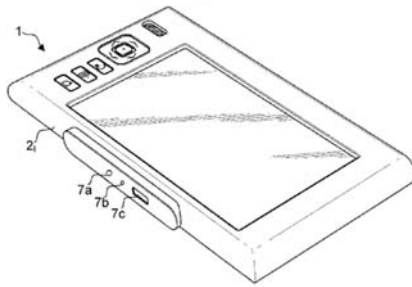


Fig 3

- Paasonen, Fig. 8

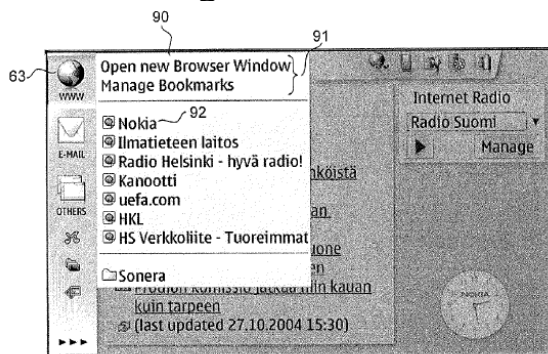


Fig 8

- Paasonen, 9:20-32 (“[A] pocket computer having a display with a user interface and a controller, the controller being adapted for execution of different utility applications, each utility application providing certain nominal functionality to a user when executed as an active application in said user interface, the pocket computer having a home application adapted for **simultaneous provision on said display of a number of limited application views** to respective ones among said utility applications, wherein **each such limited application view enables the user to access a limited part of the nominal functionality of a respective utility application without executing this utility application as an active application.**”);
- Paasonen, 9:65-10:2 (“**In one embodiment, the pocket computer is a mobile terminal accessory particularly designed for electronic browsing and messaging.**”);
- Paasonen, 11:48-65 (“...**Non-limiting examples of applications are an Internet/WWW/WAP browser application, a contacts application, a messaging application (email, SMS, MMS), a calendar application, a voice memo application, an alarm clock application, a word processing application, a spreadsheet application, a code memory application, a music**

player application, a media streaming application, and a control panel application. Some applications will be described in more detail later....”);

- **Paasonen, 13:16-52** (“...*The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be hierarchical. The lower portion 66 represents an application switcher panel with respective icons 67 for each of a plurality of launched applications. The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.”);*
- **Paasonen, 14:22-37** (“The lower portion 66 of the task navigator 60 will now be described in more detail. As already mentioned, it represents an *application switcher panel with respective icons 67 for each of a plurality of launched applications*, i.e. running applications that are executed by the controller 50. Among such running applications, *one will be active in the sense that it has control over the application area 80 on the display 3. The user 9 may conveniently use the application switcher panel 66 for switching to a desired application by tapping with the stylus 9c on the corresponding icon 67.* A help text, preferably containing the application's title and a current file name, etc, if applicable, may conveniently be presented on the display 3 next to the icon pointed at, so as to guide the user further. When the user lifts the stylus 9 c, the application corresponding to the icon pointed at will be switched to.”).

- **Paasonen, 14:60-15:5** (“As appears from the above, the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a *switch is done from a currently active application to another one, the order of the icons will be updated accordingly*. This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an audio player application that was active before the web browser application was launched*, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”)
- **See also Paasonen, 1:13-30, 3:58-63, 4:22-25, 15:19-33.**

66. Paasonen discloses methods for presenting multiple simultaneously “running applications” on a “mobile terminal.” Paasonen, 9:20-32, 9:65-10:2, 14:22-37. For example, the terminal can execute “an Internet/WWW/WAP browser application, a contacts application, a messaging application (email, SMS, MMS), a calendar application,” “a music player application, a media streaming application,” while another application is “active.” Paasonen, 11:48-65, 14:22-37. The “active application” is the one displayed on the terminal’s main display, “application area 80.” Paasonen, Fig. 8, 14:22-37, 13:25-50. Paasonen teaches controlling a “functionality” of a second application or work process, such as the browser application, while a first application, such as an “audio[/music]” player, is simultaneously playing music. Paasonen, 9:20-32, 14:60-15:5.

(b) Element [1.a]

67. **Element [1.a] recites “displaying one icon in a designated portion of a display, wherein the displayed icon corresponds to the first work process.”** This element is met by Paasonen.

68. **Paasonen discloses displaying one icon in a designated portion of a display, wherein the displayed icon corresponds to the first work process (e.g., displaying “icon” corresponding to “audio player” in “third position” of “application switcher panel”).** *See, e.g.,* Paasonen (emphasis added):

- **Paasonen, Fig. 12**

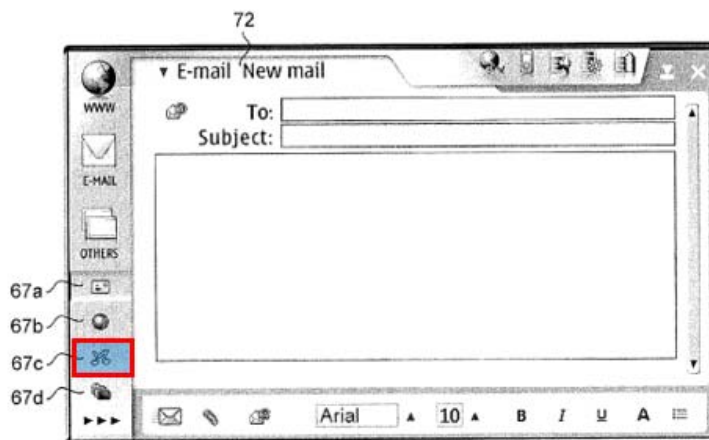


Fig 12

- **Paasonen, 14:22-37** (“*The lower portion 66 of the task navigator 60 will now be described in more detail. As already mentioned, it **represents an application switcher panel with respective icons 67 for each of a plurality of launched applications, i.e. running applications that are executed by the controller 50.** Among such running applications, **one will be active in the sense that it has control over the application area 80 on the display 3.** The user 9 may conveniently use the application switcher panel 66 for switching to a desired application by tapping with the stylus 9c on the corresponding icon 67.* A help text, preferably containing the application's title and a current file name, etc, if applicable, may conveniently be

presented on the display 3 next to the icon pointed at, so as to guide the user further. When the user lifts the stylus 9 c, the application corresponding to the icon pointed at will be switched to.”);

- **Paasonen, 14:38-48** (“[I]n the disclosed embodiment a maximum of *four different running applications will be represented by respective icons 67 in the application switcher panel 66*. The order among the icons 67 is such that *the icon for the most recently active application will be shown at the topmost position...*”);
- **Paasonen, 15:6-18** (“Now, *when the user 9 invokes the messaging application by selecting the menu item 93 in the afore-described task-oriented menu 90, the messaging application becomes active and its icon takes the topmost position 67 a*, as seen in FIG. 12. At the same time, *the existing icons 67 a-c of FIG. 11 are shifted one vertical position downwards, so that the web browser icon (formerly at 67 a) takes the second position at 67 b, the audio player icon moves to the third position 67 c, and the file manager icon goes to the lowermost position 67 d*. The formerly shown image viewer icon disappears from the application switcher panel 66, but the image viewer application is still running.”).
- **See also Paasonen Figs. 7-9, 13, 8:58-9:4, 14:49-59, 14:60-15:5, 15:19-33.**

69. Paasonen discloses the mobile terminal’s “display” includes an “application switcher panel 66,” shown on the left side on the display in Figure 12. Paasonen, Fig. 12, 14:22-37. The switcher panel displays application “icons” corresponding to “different running applications” such that the “most recently active application” is “shown at the topmost position.” Paasonen, 14:38-48. The icons are ordered according to how recently the corresponding application was active, such that the most recently active application is shown in the “topmost position 67a,” the next most recent application is represented in the second position, and so forth. Paasonen, 14:38-48, 14:60-15:5, 15:6-18. For example, the

“audio player” icon is displayed in “third position 67c” in Figure 12, indicating it was the third most recently active application. Paasonen, 15:6-18.

(c) Element [1.b]

70. **Element [1.b]** recites “if the displayed icon corresponding to the first work process is selected, displaying a screen of the first work process in the display.” This element is met by Paasonen.

71. **Paasonen discloses if the displayed icon corresponding to the first work process is selected** (*e.g.*, if “the corresponding icon” for the “audio player” is “tapp[ed]”), displaying a screen of the first work process in the display (*e.g.*, “switching to” the “audio player” in “application area 80”). *See, e.g.*, Paasonen (emphasis added):

• **Paasonen, Fig. 7**

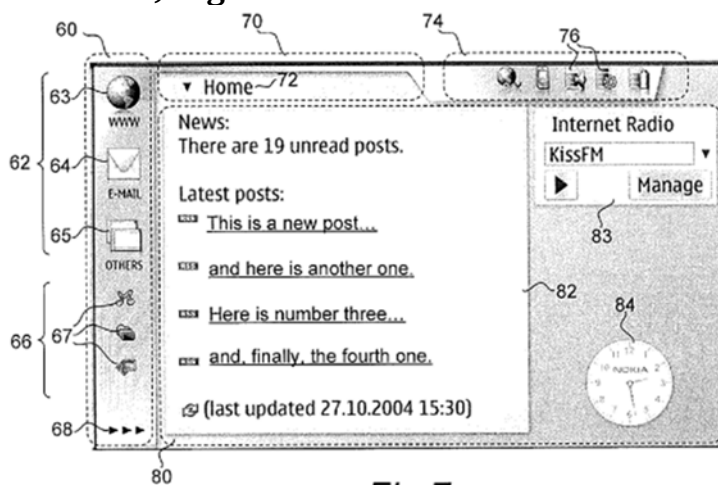


Fig 7

- **Paasonen, Fig. 12**

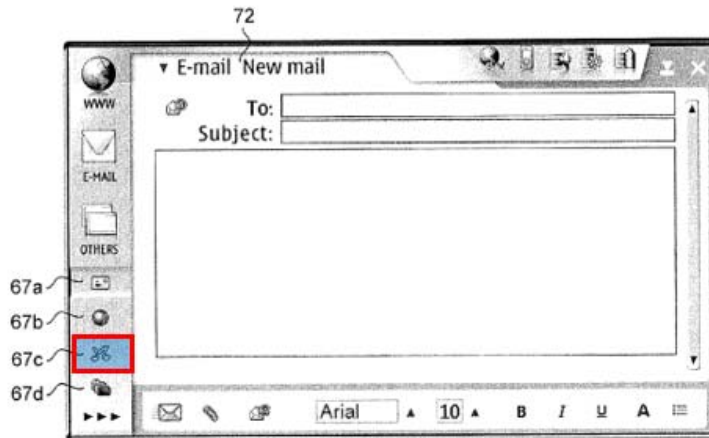


Fig 12

- **Paasonen, 12:56-13:7** (“With reference to FIG. 7, *the display screen layout of the display 3 is divided into four main areas: a task navigator 60, a title area 70, a status indicator area 74 and an application area 80.* The *application area 80 is used by a currently active application to present whatever information (content) is relevant* and also to provide user interface controls such as click buttons, scrollable list, check boxes, radio buttons, hyper links, etc, which allow the user to interact with the currently active application by way of the stylus 9c. *One example of how a currently active application, in the form of a web browser, uses the application area 80 in this manner is shown in FIG. 9.... In addition, as seen in FIG. 10, by tapping in the title area 70, the user may access an application menu 73 of the currently active application.*”);
- **Paasonen, 14:22-37** (“*The lower portion 66 of the task navigator 60 will now be described in more detail. As already mentioned, it represents an application switcher panel with respective icons 67 for each of a plurality of launched applications, i.e. running applications that are executed by the controller 50. Among such running applications, one will be active in the sense that it has control over the application area 80 on the display 3. The user 9 may conveniently use the application switcher panel 66 for switching to a desired application by tapping with the stylus 9c on the corresponding icon 67. A help text, preferably containing the application's title and a current file name, etc, if applicable, may conveniently be presented on the display 3 next to the icon pointed at, so as to guide the user further. When the user lifts the stylus 9 c, the application corresponding to the icon pointed at will be switched to.*”);

- **Paasonen, 14:60-15:5** (“As appears from the above, *the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly.* This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an audio player application that was active before the web browser application was launched*, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”).
- **See also Paasonen Figs 8-13, 11:48-65, 14:49-59, 15:19-33.**

72. See my discussion of element [1.a]. As explained for element [1.a] above, Paasonen discloses displaying work process icons in a designated portion of a display. See [1.a]. Paasonen further discloses switching to a selected application by “tapping” on its corresponding icon in the application switcher panel. Paasonen, 14:22-37. The “active application” is then displayed in “application area 80” shown in Figure 7. Paasonen, Fig. 7, 12:56-13:7. For example, selecting the icon corresponding to the “audio player application” (which was in the third position in the application switcher panel) launches the audio player in the application area. Paasonen, 14:60-15:5.

(d) Element [1.c]

73. **Element [1.c]** recites “in response to a first command for sequentially displaying icons, performing sequential display of icons such that an icon corresponding to the second work process is displayed, wherein the

icon corresponding to the second work process is displayed in the designated portion of the display.” This element is met by Paasonen.

74. Paasonen discloses in response to a first command for sequentially displaying icons (e.g., command to launch a different application, such as a “bookmark manager”), performing sequential display of icons such that an icon corresponding to the second work process is displayed (e.g., “order of the icons” is “updated accordingly” when switching “from a currently active application to another one” by “shifting” icons “one vertical position downwards”), wherein the icon corresponding to the second work process is displayed in the designated portion of the display (e.g., the “web browser icon” is displayed in “third position 67c” of “application switcher panel 66” in Fig. 13). See, e.g., Paasonen (emphasis added):

- Paasonen, Fig. 5

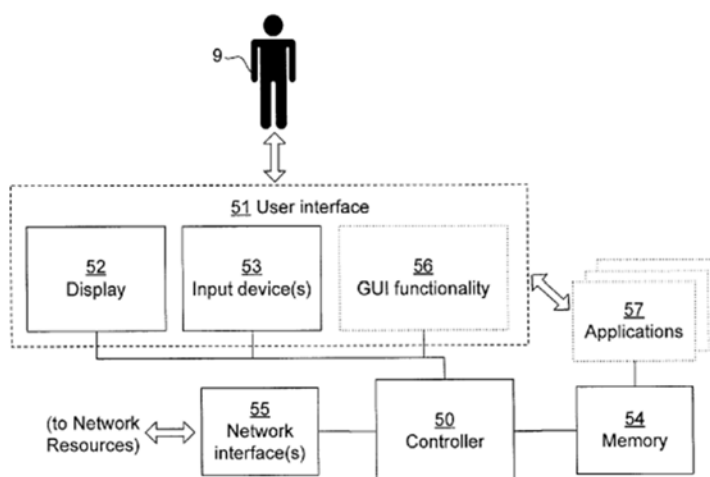


Fig 5

- Paasonen, Fig. 12

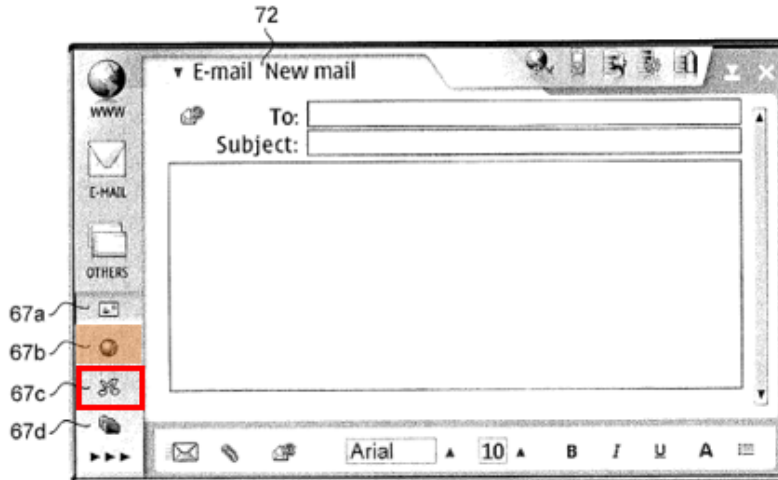


Fig 12

- Paasonen, Fig. 13

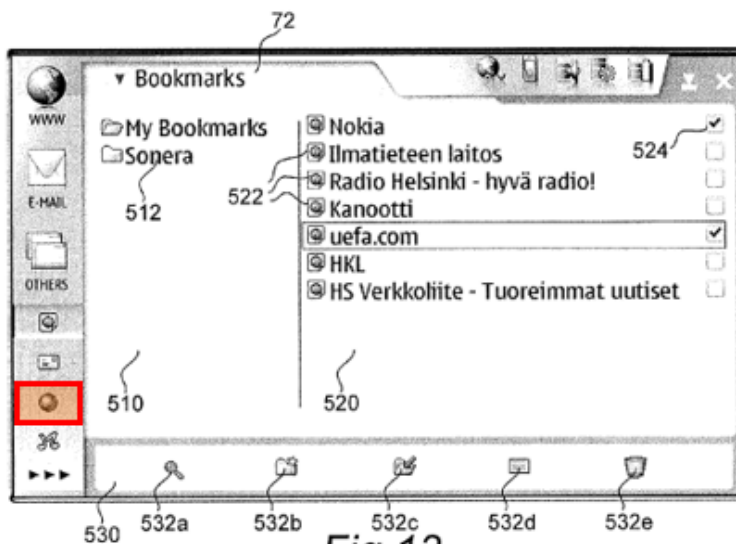


Fig 13

- Paasonen, 9:5-16 (“The graphical user interface may be further adapted, upon launching of a new application, to insert an icon associated with said new application at said first position in the application switcher panel while shifting the positions of existing icons in the application switcher panel by one position backwards.
In one embodiment, *only a predetermined maximum number of positions for icons may be allowed* in said application switcher panel wherein, for *an icon that has been shifted out from the application switcher panel*, the

application associated therewith may be activated through selection of a menu item in a menu on said display.”);

- **Paasonen, 13:16-52** (“...*The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91*, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be hierarchical. The lower portion 66 represents an application switcher panel with respective icons 67 for each of a plurality of launched applications. *The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.*”);
- **Paasonen, 14:38-48** (“[I]n the disclosed embodiment a maximum of *four different running applications will be represented by respective icons 67 in the application switcher panel 66*. The order among the icons 67 is such that *the icon for the most recently active application will be shown at the topmost position...*”);
- **Paasonen 14:60-15:5** (“As appears from the above, *the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly*. This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an audio player application that was active before the web browser*

application was launched, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”);

- **Paasonen, 15:6-18** (“Now, *when the user 9 invokes the messaging application by selecting the menu item 93 in the afore-described task-oriented menu 90, the messaging application becomes active and its icon takes the topmost position 67 a*, as seen in FIG. 12. At the same time, *the existing icons 67 a-c of FIG. 11 are shifted one vertical position downwards, so that the web browser icon (formerly at 67 a) takes the second position at 67 b, the audio player icon moves to the third position 67 c, and the file manager icon goes to the lowermost position 67 d*. The formerly shown image viewer icon disappears from the application switcher panel 66, but the image viewer application is still running.”).
- **See also Paasonen Figs. 6, 7-9, 8:58-9:4, 11:22-31, 13:38-45, 14:22-29.**

75. See my discussion of element [1.a]. As explained for element [1.a] above, Paasonen discloses displaying work process icons in “the designated portion of the display.” See [1.a]. In addition, Paasonen explains that icons in the “application switcher panel” are ordered vertically from most- to least-recently active. Paasonen, 14:38-48, 14:60-15:5. The user “select[s]” icons in the switcher panel to launch, or “switch[],” from one “currently active application to another.” Paasonen, 14:60-15:5, 15:6-18. When “switch[ing]” from one application to another, the order of icons in the application switcher panel is updated accordingly such that “icons 67a-c” shown in Figure 12 are “shifted one vertical position downwards.” Paasonen, Figs. 12–13, 9:5-16, 13:16-52, 14:60-15:18. For example, the user’s selection of a new application such as a “bookmark manager” application while the web browser application was the second-most-recently

activated application—meaning that the corresponding icon for the web browser was located in second position 67b of the application switcher panel—causes the web browser icon to move from the second position 67b to third position 67c. Paasonen, Figs. 12–13.

76. Indeed, this disclosure of this element is similar to an embodiment of incrementally displaying icons that was described by the Patent Owner during prosecution of the parent to the '631 patent. The '631 patent claims priority to U.S. Patent No. 9,122,518. I understand that a copy of the prosecution history of U.S. App. No 11/495,863, which matured into the '518 Patent, has been designated as Exhibit 1030 (“863-App. File History”). During prosecution of that patent, the Patent Owner stated that ordering work process icons by how recently they were active discloses this limitation:

Further, Nokia discloses navigation of menu items for selection, prior to execution of an application or process corresponding to the respective menu item. Nokia, page 19. However, once a menu item is selected for execution, Nokia does not have any further involvement in the actual display of “icons” corresponding to the “work process performed”. Accordingly, the display of mere menu items for selection does not equate to the “sequentially displaying of icons in turn corresponding to the respective work process of the mobile terminal, wherein sequentially displaying icons comprises displaying the icons in the order of the work process performed”.

See 863-App. File History, 182–91. The Patent Owner also amended the pending claims in that parent patent to expressly recite that “sequentially displaying icons comprises displaying the icons in order of the work process performed.” 863-App. File History, 183. And the Examiner also separately stated that the generation and display of icons “in a sequential order” according to an “activation order”—i.e., the order in which they were most recently opened—is a form of sequential display. 863-App. File History, 214–23, 373–89. Paasonen teaches this type of sequential display: the application switcher sequentially displays icons by placing the icons in what the Examiner described as an “activation order,” from most- to least-recently opened, which is also the order of the work process performed. 863-App. File History, 214–23, 373–89.

(e) Element [1.d]

77. **Element [1.d] recites “in response to a second command for changing to the second work process, replacing the screen of the first work process with a screen of the second work process.”** This element is met by Paasonen.

78. **Paasonen discloses in response to a second command for changing to the second work process (e.g., command for “switching to a desired application,” “web browser”), replacing the screen of the first work process with a screen of the second work process (e.g., changing “active” application in**

“application area 80” of the “display” to “web browser”). See, e.g., Paasonen (emphasis added):

- **Paasonen, Fig. 9**



Fig 9

- **Paasonen, 12:56-13:7** (“With reference to FIG. 7, *the display screen layout of the display 3 is divided into four main areas: a task navigator 60, a title area 70, a status indicator area 74 and an application area 80.* The *application area 80 is used by a currently active application to present whatever information (content) is relevant* and also to provide user interface controls such as click buttons, scrollable list, check boxes, radio buttons, hyper links, etc, which allow the user to interact with the currently active application by way of the stylus 9c. *One example of how a currently active application, in the form of a web browser, uses the application area 80 in this manner is shown in FIG. 9.... In addition, as seen in FIG. 10, by tapping in the title area 70, the user may access an application menu 73 of the currently active application.*”);
- **Paasonen, 14:22-37** (“*The lower portion 66 of the task navigator 60 will now be described in more detail. As already mentioned, it represents an application switcher panel with respective icons 67 for each of a plurality of launched applications, i.e. running applications that are executed by the controller 50. Among such running applications, one will be active in the sense that it has control over the application area 80 on the display 3. The user 9 may conveniently use the application switcher panel 66 for switching to a desired application by tapping with the stylus 9c on the corresponding icon 67. A help text, preferably containing the application's*

title and a current file name, etc, if applicable, may conveniently be presented on the display 3 next to the icon pointed at, so as to guide the user further. When the user lifts the stylus 9 c, the application corresponding to the icon pointed at will be switched to.”);

- **Paasonen, 14:60-15:5** (“As appears from the above, *the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly.* This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an audio player application that was active before the web browser application was launched*, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”).
- **See also Paasonen Figs. 5-8, 11-12; 11:22-31, 11:48-65, 14:49-59, 15:6-18, 15:19-33.**

79. See my discussion of element [1.b]. As explained for element [1.b] above, Paasonen discloses if an icon corresponding to a work process is selected, displaying a screen of the work process. See [1.b]. Paasonen further discloses “switching” from one application, such as the music player application, to another application, such as the “web browser” application by selecting “the corresponding icon” in the switcher panel. Paasonen, 14:22-37. Doing so causes the “web browser” application to be displayed in “application area 80.” Paasonen, 12:56-13:7.

(f) Element [1.e]

80. **Element [1.e] recites “controlling the multitasking of the second work process and the first work process.”** This element is met by Paasonen.

81. Paasonen discloses controlling the multitasking of the second work process and the first work process (e.g., “switch” between “running applications” including the first and second work processes (see [1.pre])). See, e.g., Paasonen (emphasis added):

- Paasonen, 14:22-29 (“*The lower portion 66 of the task navigator 60 will now be described in more detail. As already mentioned, it **represents an application switcher panel with respective icons 67 for each of a plurality of launched applications, i.e. running applications that are executed by the controller 50.** Among such running applications, **one will be active in the sense that it has control over the application area 80 on the display 3.** The user 9 may conveniently use the application switcher panel 66 for switching to a desired application by tapping with the stylus 9c on the corresponding icon 67. A help text, preferably containing the application's title and a current file name, etc, if applicable, may conveniently be presented on the display 3 next to the icon pointed at, so as to guide the user further. When the user lifts the stylus 9 c, the application corresponding to the icon pointed at will be switched to.”);*
- Paasonen, 14:60-15:5 (“As appears from the above, *the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly.* This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an audio player application that was active before the web browser application was launched*, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”).
- See also Paasonen 3:58-63, 4:22-25, 9:20-32, 13:16-52, 15:19-33.

82. See my discussion of elements [1.pre], [1.b], and [1.d]. As explained for element [1.pre] above, Paasonen discloses a first work process and a second

work process in a multitasking mobile terminal. *See* [1.pre]. As I further discuss for elements [1.b] and [1.d], Paasonen discloses first and second work processes. Paasonen teaches running multiple applications, including the “web browser” and “audio player” applications, simultaneously, and using the application switcher panel to “switch” between these running applications. Paasonen, 14:22-37, 14:60-15:5. *See* [1.b], [1.d].

(g) Element [1.f]

83. **Element [1.f] recites “in response to a third command for activating a control menu for controlling a predetermined function of the first work process, activating the control menu for controlling the predetermined function of the first work process.”** This element is met by Paasonen.

84. **Paasonen discloses in response to a third command for activating a control menu for controlling a predetermined function of the first work process** (*e.g.*, invoking “the home application” to view “application views”; activating “menu 90” for “audio player”), **activating the control menu for controlling the predetermined function of the first work process** (*e.g.*, “application view” “provide[s] access” to a “limited part” of the “functionality nominally provided by such ... application”; preset “menu” of “other tasks”). *See, e.g.*, Paasonen (emphasis added):

- Paasonen, Figs. 6–8

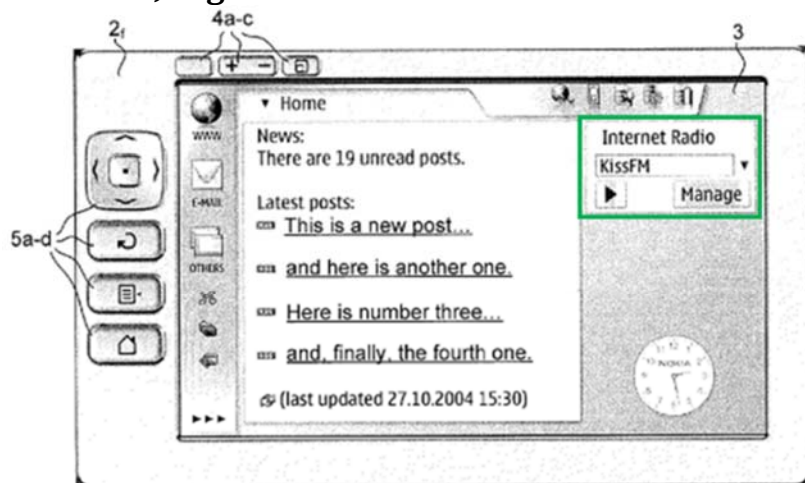


Fig 6

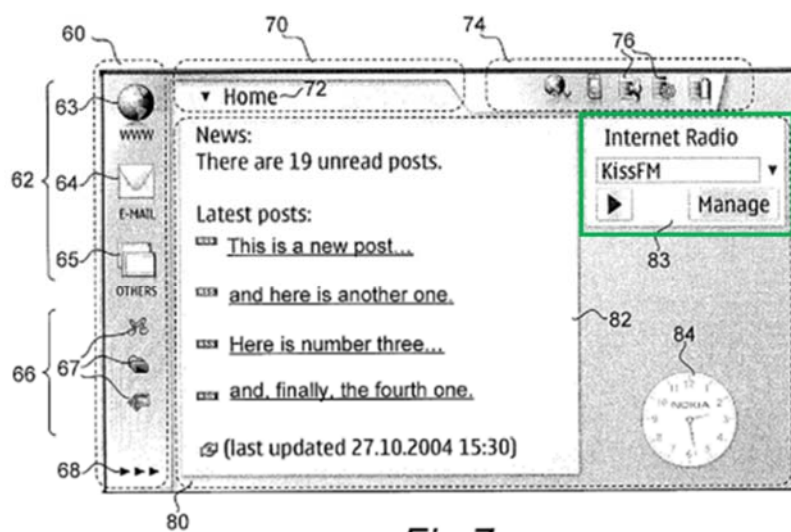


Fig 7

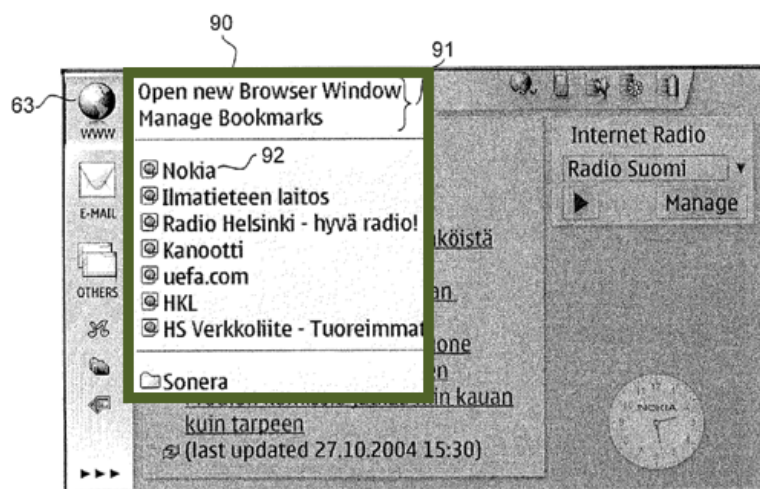


Fig 8

- **Paasonen, Fig. 11**

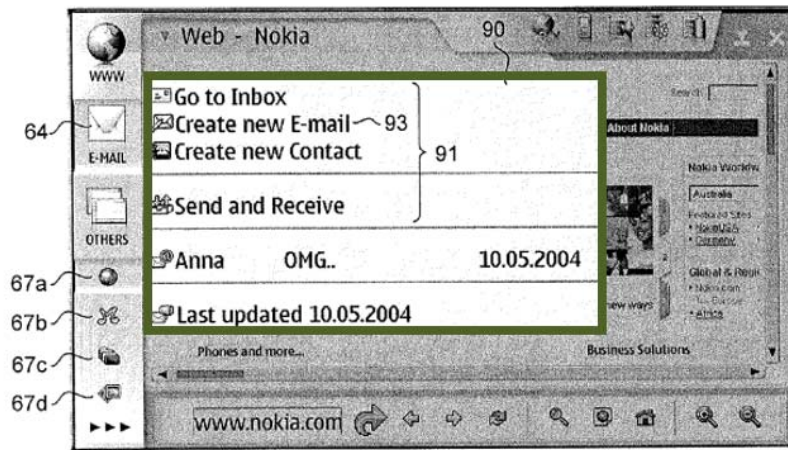


Fig 11

- **Paasonen, 9:20-32** (“[A] pocket computer having a display with a user interface and a controller, the controller being adapted for execution of different utility applications, each utility application providing certain nominal functionality to a user when executed as an active application in said user interface, the pocket computer having a home application adapted for *simultaneous provision on said display of a number of limited application views* to respective ones among said utility applications, wherein *each such limited application view enables the user to access a limited part of the nominal functionality of a respective utility application without executing this utility application as an active application.*”);
- **Paasonen, 11:48-65** (“...*Non-limiting examples of applications are an Internet/WWW/WAP browser application, a contacts application, a messaging application (email, SMS, MMS), a calendar application, a voice memo application, an alarm clock application, a word processing application, a spreadsheet application, a code memory application, a music player application, a media streaming application, and a control panel application.* Some applications will be described in more detail later...”);
- **Paasonen, 13:16-52** (“...*The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be*

hierarchical. The lower portion 66 represents an application switcher panel with respective icons 67 for each of a plurality of launched applications.

The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.”);

- **Paasonen, 14:14-17 (“Selection of the third icon 65 will cause presentation of a menu 90 with links to other tasks that are available, e.g. the various ones among the applications 57 that are not related to information browsing or electronic messaging.”);**
- **Paasonen, 14:60-15:5 (“As appears from the above, the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly. This is shown in FIGS. 11 and 12. In FIG. 11, the web browser application is active and is thus represented by the topmost icon 67 a. The second icon 67 b represents an audio player application that was active before the web browser application was launched, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”);**
- **Paasonen, 15:61-16:40 (“During ongoing use of the pocket computer 1, irrespective of whatever other application that is currently active, the user 9 may always return to the home application by pressing the home key 5d on the front surface 2f of the apparatus housing 2. Another way of invoking the home application is through the application switcher menu button 68, as has been described above.**

As seen in FIG. 7, in this embodiment ***the home application contains three application views 82, 83 and 84 on the display 3. Each application view is a downscaled version of the application view of another application 57.***

Thus, among all the functionality nominally provided by such another application 57, ***the application view in the home application will only provide access to limited parts thereof.*** For instance, application view 82 in FIG. 7 represents a news application (e.g. Usenet news) and provides a limited view of this application by displaying the number of unread posts together with a few of the latest posts. Tapping on any of these latest posts will cause presentation of the contents of the post in question. If the user wants to access the complete functionality of the news application, he may switch to this application through e.g. the application switcher menu button 68 (as described above), or the “Others” icon 65 in the upper part 62 of the task navigator 60. In another embodiment, tapping on a post in the application view 82 may directly cause launching (if not already running) of or switching to the news application.

The application view 83 represents an Internet radio application and gives a limited view of its functionality. By tapping on a “Manage” button therein, the user may invoke the actual Internet radio application to access its entire functionality. The application view 84 represents a Clock application. ***The interaction between such a limited application view 82, 83, 84 and the actual application*** it represents may be implemented using push technique, as is readily realized by a skilled person.

In one embodiment, the ***user may configure which application views to include*** in the home application, and some particulars of them.

Using only limited resources in terms of memory, CPU load and display screen space, the home application gives the user 9 a very convenient overlook view of certain applications that he probably likes to access frequently.”).

- ***See also Paasonen Fig. 5, 9:20-32, 11:22-31, 11:48-65, 14:60-15:5, 16:34-36.***

85. See my discussion of elements [1.pre] and [1.b]. As explained for element [1.pre] above, Paasonen discloses controlling a predetermined function of a first work process. See [1.pre]. As I further discuss for element [1.b] above, Paasonen discloses activating and displaying a screen of the first work process.

See [1.b]. In addition, Paasonen discloses two features that teach a third command for activating a control menu for controlling a predetermined function of a work process.

86. First, as explained in §IX.A.1, Paasonen’s “home application” includes “limited application views” that provide “limited” access to application functions for different applications. Paasonen, 9:20-32, 15:61-16:40. For example, “application view 83” shown in Figures 6–7, provides access to certain functions for a media streaming application, “Internet radio application,” such as “stop” and “play” buttons that control the application’s stop and play functions. Paasonen, Figs. 6–7, 15:61-16:40. The home application is invoked by selecting “the application switcher menu button 68.” Paasonen, 15:61-16:3. Paasonen’s “limited application views” are user “configur[able]” such that a user can configure which application to access through the limited view. Paasonen, 15:61-16:40. For example, the “music player application” and “media streaming application”, discussed for [1.pre], may be accessed through the limited application view. Paasonen, 11:48-65, 15:61-16:40 (“As seen in FIG. 7, in this embodiment the home application contains three application views 82, 83 and 84 on the display 3. Each application view is a downscaled version of the **application view of another application** 57....”). Because the application views are user-configurable, Paasonen’s “application view” not only renders obvious, but affirmatively

discloses a control menu for controlling a predetermined function of the first work process.

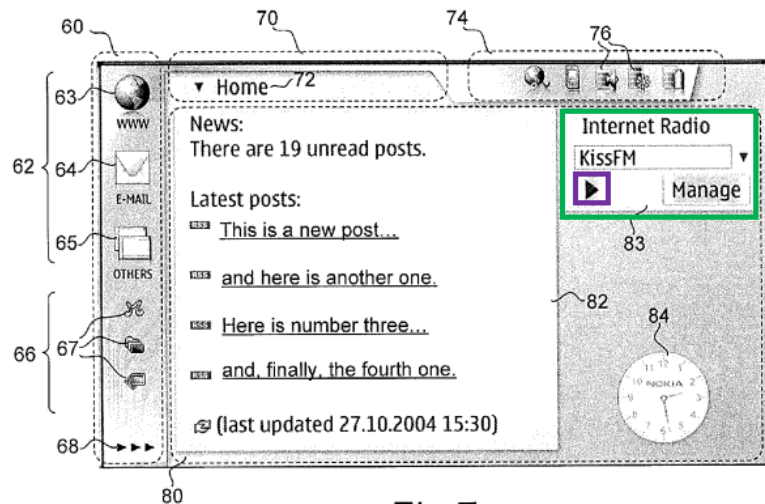
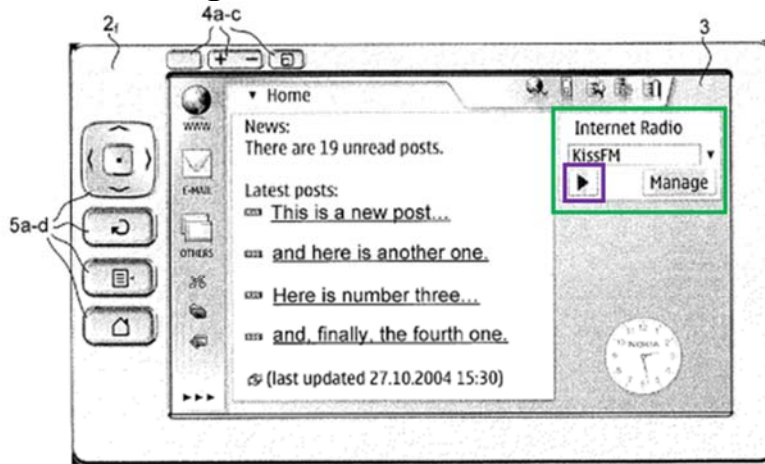
87. Second, the application switcher panel includes a “third icon 65,” shown in Figure 7, that is selected to open a “menu 90” containing “task-oriented menu items” for available applications. Paasonen, Figs. 7–8, 11, 13:16-52, 14:14-17. See §IX.A.1 The task-oriented menu provides users with application-specific controls. Paasonen, 13:16-52. For example, a context-sensitive menu for Paasonen’s audio player application that permits a user to control audio playback with a play, pause or stop option. See Paasonen, Figs. 8, 11. As shown in Figure 6, Paasonen’s “limited application view” and “menu 90” are provided in a single embodiment of Paasonen’s “home” display screen—menu 90 is accessed from the application switcher panel on the left side and the application view is on the right side of the screen. Paasonen, 14:14-17, 15:61-16:40, Fig. 6. They provide two separate means for users to access features of the same applications, e.g. the music player application and media streaming application, from the home screen. Thus, opening “menu 90” not only renders obvious, but affirmatively discloses, activating a menu for controlling predetermined functions of the audio/music player applications.

(h) Element [1.g]

88. **Element [1.g] recites “in response to a fourth command for controlling the predetermined function of the first work process through the activated control menu, controlling the predetermined function of the first work process through the activated control menu.”** Paasonen discloses this element.

89. **Paasonen discloses in response to a fourth command for controlling the predetermined function of the first work process through the activated control menu** (*e.g.*, pressing play in “limited application view”; selecting play task in “menu 90”), **controlling the predetermined function of the first work process through the activated control menu** (*e.g.*, to access “limited parts” of the “audio player” functionality by playing music). *See, e.g.*, Paasonen (emphasis added):

- **Paasonen, Figs. 6–7**



- **Paasonen, 14:14-17** (“*Selection of the third icon 65 will cause presentation of a menu 90 with links to other tasks that are available, e.g. the various ones among the applications 57 that are not related to information browsing or electronic messaging.*”);
- **Paasonen, 14:60-15:5** (“As appears from the above, *the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly.* This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an*

audio player application that was active before the web browser application was launched, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”).

- **Paasonen, 15:61-16:40** (“During ongoing use of the pocket computer 1, irrespective of whatever other application that is currently active, the **user 9 may always return to the home application by pressing the home key 5d on the front surface 2f of the apparatus housing 2**. Another way of **invoking the home application is through the application switcher menu button 68**, as has been described above.

As seen in FIG. 7, in this embodiment **the home application contains three application views 82, 83 and 84 on the display 3. Each application view is a downscaled version of the application view of another application 57.**

Thus, among all the functionality nominally provided by such another application 57, **the application view in the home application will only provide access to limited parts thereof.** For instance, application view 82 in FIG. 7 represents a news application (e.g. Usenet news) and provides a limited view of this application by displaying the number of unread posts together with a few of the latest posts. Tapping on any of these latest posts will cause presentation of the contents of the post in question. If the user wants to access the complete functionality of the news application, he may switch to this application through e.g. the application switcher menu button 68 (as described above), or the “Others” icon 65 in the upper part 62 of the task navigator 60. In another embodiment, tapping on a post in the application view 82 may directly cause launching (if not already running) of or switching to the news application.

The application view 83 represents an Internet radio application and gives a limited view of its functionality. By tapping on a “Manage” button therein, the user may invoke the actual Internet radio application to access its entire functionality. The application view 84 represents a Clock application. **The interaction between such a limited application view 82, 83, 84 and the actual application** it represents may be implemented using push technique, as is readily realized by a skilled person.

In one embodiment, the **user may configure which application views to include** in the home application, and some particulars of them.

Using only limited resources in terms of memory, CPU load and display screen space, the home application gives the user 9 a very convenient

overlook view of certain applications that he probably likes to access frequently.”).

-
- **See also Paasonen Fig. 5, 9:20-32, 9:65-10:2, 11:21-31.**

90. See my discussion of element [1.f]. As explained for element [1.f] above, Paasonen discloses an activated control menu for controlling a predetermined function of the first work process. See [1.f]. In Paasonen’s “limited application view” feature, a user invokes the play button, by selecting or tapping on it, to control the audio player application’s stop/play function. See Paasonen, Figs. 6–8, 15:61-16:40. In Paasonen’s “context-specific menu” feature, a user may invoke similar application-specific tasks, including stop/play, to control the audio player’s functions. See Paasonen, 14:14-17.

(i) Claim [2]

91. **Claim [2] recites “The method of claim 1, wherein the controlling the predetermined function of the first work process comprises: generating a control signal for controlling the predetermined function of the first work process in response to the fourth command; and controlling the predetermined function of the first work process in response to the generated control signal.”** This Claim is met by Paasonen.

92. **Paasonen further discloses that the controlling the predetermined function of the first work process comprises: generating a control signal for**

controlling the predetermined function of the first work process in response to the fourth command (e.g., see [1.f]-[1.g]; “controller” uses “drivers” and generates control signal for controlling “audio player”); and controlling the predetermined function of the first work process in response to the generated control signal (e.g., “controller” controls “audio player” application in response to control signal). See, e.g., Paasonen (emphasis added):

- **Paasonen, Figure 5**

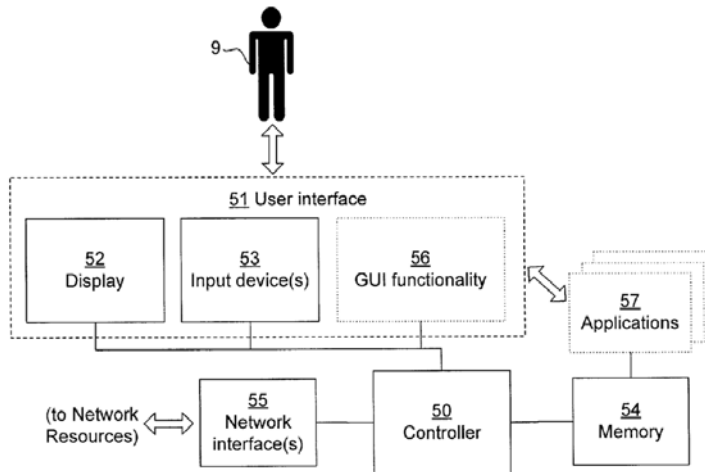


Fig 5

- **Paasonen, 11:32-47** (“As seen in FIG. 5, the pocket computer 1 also *has a controller 50 with associated memory 54. The controller is responsible for the overall operation of the pocket computer 1 and may be implemented by any commercially available CPU* (Central Processing Unit), DSP (Digital Signal Processor) or any other electronic programmable logic device. The associated memory may be internal and/or external to the controller 50 and may be RAM memory, ROM memory, EEPROM memory, flash memory, hard disk, or any combination thereof. The memory 54 is used for various purposes by the controller 50, one of them being for storing data and program instructions for various pieces of software in the pocket computer 1. *The software may include a real-time operating system, drivers e.g. for the user interface 51, as well as various applications 57.*”).

- **See also Paasonen 9:20-32, 11:22-31, 16:30-33.**

93. See my discussion of elements [1.pre] and [1.f]–[1.g]. As explained for elements [1.pre] and [1.f]–[1.g] above, Paasonen discloses controlling a predetermined function of a first work process (which is an audio player, as I discussed for [1.pre]) in response to a fourth command. See [1.pre], [1.f]–[1.g]. Paasonen further teaches a “controller 50” that is “responsible for the overall operation” of the mobile terminal. Paasonen, 11:32-47. Mobile terminal applications are executed by a controller coupled to user interface 51 that accepts user input. See Paasonen, Fig. 5. A person of ordinary skill in the art would have understood that controllers execute user commands by accepting user input signals and generating control signals to execute the commands. For example, Paasonen’s controller is coupled to “memory 54” which stores “data and program instructions” for applications stored on the terminal. Paasonen, Fig. 5, 11:32-47. When the controller receives an input signal from user interface 51, for example, to control Paasonen’s audio player application, the controller issues control signals to access audio application program data and have the user interface drivers play or stop audio playback. See Paasonen, 11:32-47. When a controller interfaces with physical input or output devices, such as speakers, the controller uses “drivers” to communicate with the hardware to, for example, output audio such as music. See Paasonen, 11:32-47. Such controller functions of generating control signals was

well known in the art. *See, e.g.*, Koenck (Ex. 1013), [0104] (explaining that “[]controller . . . communicat[es] data and control signals” to other modules of a portable terminal); Loke (Ex. 1014), [0072] (describing an “automatic gain control signal AGC from a central controller” in a mobile phone). Indeed, the ’631 Patent discussion of the prior art describes a mobile terminal chipset that issues “control signals” to control device functions. ’631 patent, 1:46-51, 2:26-28.

(j) Claim [3]

94. **Claim 3 recites “The method of claim 1, further comprising: in response to a fifth command for deactivating the activated control menu, deactivating the activated control menu.”** This Claim is met by Paasonen.

95. **Paasonen further discloses in response to a fifth command for deactivating the activated control menu** (e.g., a command for “opening a new browser window;” “switching” to a new application), **deactivating the activated control menu** (see [1.f]; navigating away from the home application or to another application deactivates the “limited application view” or “menu 90”). *See, e.g.*, Paasonen (emphasis added):

- **Paasonen, Figs. 8–9**

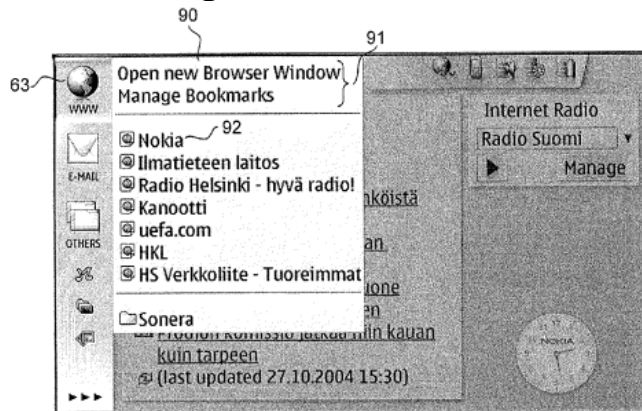


Fig 8

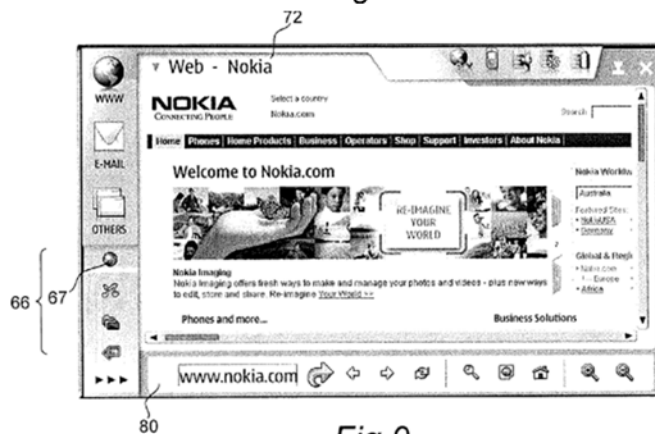


Fig 9

- **Paasonen, 13:38-45** (“*More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application* (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein.”);
- **Paasonen, 14:30-37** (“*The user 9 may conveniently use the application switcher panel 66 for switching to a desired application by tapping with the stylus 9c on the corresponding icon 67. A help text, preferably containing the application's title and a current file name, etc, if applicable, may conveniently be presented on the display 3 next to the icon pointed at, so as to guide the user further. When the user lifts the stylus 9 c, the application corresponding to the icon pointed at will be switched to.*”).
- **See also Paasonen Fig. 7; 11:4-11, 15:19-33, 15:61-16:40.**

96. See my discussion of elements [1.f]–[1.g]. As explained for [1.f]–[1.g] above, Paasonen discloses a third command for activating a control menu. Paasonen further teaches deactivating the control menus by issuing a command to “open a new browser window” or “switch[]” to a new application. Paasonen, 13:38-45, 14:30-37. See [1.f]–[1.g]. In view of the two control menu features discussed for element [1.f], a user deactivates Paasonen’s limited application view with audio player functions by exiting the home application and deactivates context-specific “menu 90” simply by navigating away from it. Paasonen, 13:38-45, 14:30-37. For example, selecting a new application or opening a new browser window navigates away from and displaces the context-specific menu. A person of ordinary skill in the art would have found it obvious that such commands close control menus because the process of navigating away from these control menus requires the menus to no longer be displayed—i.e., for the control menus to be deactivated once the user is no longer interacting with the control menu.

(k) Claim 4

97. **Claim 4 recites “The method of claim 1, wherein the first work process is an MP3 play process.”** This Claim is met by Paasonen.

98. **Paasonen further discloses that the first work process is an MP3 play process** (see [1.a]; first work process is a “music” or “audio player”). See, e.g., Paasonen (emphasis added):

- **Paasonen, 11:48-65** (“...*Non-limiting examples of applications are an Internet/WWW/WAP browser application, a contacts application, a messaging application (email, SMS, MMS), a calendar application, a voice memo application, an alarm clock application, a word processing application, a spreadsheet application, a code memory application, a music player application, a media streaming application, and a control panel application. Some applications will be described in more detail later...*”);
- **Paasonen, 14:60-15:5** (“As appears from the above, *the vertical order of the application switcher icons from top to bottom represents a historical order in which the four most recently used applications have been active. When a switch is done from a currently active application to another one, the order of the icons will be updated accordingly.* This is shown in FIGS. 11 and 12. In FIG. 11, *the web browser application is active* and is thus represented by the topmost icon 67 a. The second *icon 67 b represents an audio player application that was active before the web browser application was launched*, whereas the third and fourth icons 67 c and 67 d represent a file manager application and an image viewer application, respectively, that were active before that.”).
- **See also Paasonen 16:24-29.**

99. See my discussion of element [1.a]. As explained for [1.a] above, Paasonen discloses the first work process. See [1.a]. Paasonen discloses a “music” or “audio player application” installed on its mobile terminal. Paasonen, 11:48-65, 14:60-15:5. A person of ordinary skill in the art would have understood Paasonen to teach its music player application is an MP3 play process. A person of ordinary skill in the art would have understood and recognized the well-understood and

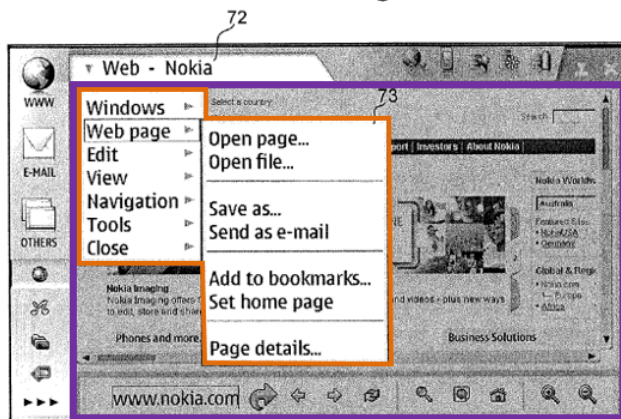
well-known fact that music players in the relevant time period would have played audio file formats including MP3 files to obtain the benefit of compatibility with a common music file format. The MP3 format was already commonly used at that time, particularly because the compressed MP3 file format allowed users to store a large library of music files for playback. *E.g.*, Janik (Ex. 1015), [0147] (“A common type of audio file format is the MP3 (MPEG layer 3) format.”); Marriott (Ex. 1016), [0071] (“[T]he media player 100 is a pocket sized hand held MP3 music player that allows a user to store a large collection of music”). Disclosures of music players in this time period therefore would have not only rendered obvious but affirmatively disclosed the use of the MP3 play processes in order to play MP3 music files. Indeed, the ’631 patent recognizes that an MP3 player was known in the prior art. ’631 patent, Fig. 1, Fig. 3, 1:38-41, 1:46-51, 2:6-28.

(I) Claim [5]

100. **Claim 5 recites “The method of claim 1, wherein an item of the first work process is displayed relatively brighter or clearer than the screen of the first work process.”** This element is met by Paasonen.

101. **Paasonen further discloses that an item of the first work process is displayed relatively brighter or clearer than the screen of the first work process** (*e.g.*, “application menu 73 of the currently active application” is brighter and clearer than “application area 80”). *See, e.g.*, Paasonen (emphasis added):

- Paasonen, Figs. 9–10



- Paasonen, 12:56-13:7 (“With reference to FIG. 7, *the display screen layout of the display 3 is divided into four main areas: a task navigator 60, a title area 70, a status indicator area 74 and an application area 80. The application area 80 is used by a currently active application to present whatever information (content) is relevant and also to provide user interface controls such as click buttons, scrollable list, check boxes, radio buttons, hyper links, etc, which allow the user to interact with the currently active application by way of the stylus 9c. One example of how a currently active application, in the form of a web browser, uses the application area 80 in this manner is shown in FIG. 9.... In addition, as seen in FIG. 10, by tapping in the title area 70, the user may access an application menu 73 of the currently active application.*”).

102. See my discussion of elements [1.a]–[1.b]. As explained for [1.a] above, Paasonen discloses items (e.g., icons) of a first work process. See [1.a]. As I further discuss for element [1.b] above, Paasonen discloses displaying a screen of the first work process. See [1.b]. In addition, Paasonen discloses “an application menu 73 of the currently active application” displayed brighter than the active application screen in application area 80, as shown in Figure 10. Paasonen, Figs. 9–10, 12:56-13:7. For example, tapping in “title area 70” of the audio player’s application screen causes a pop-up audio player application menu to be displayed brighter than the application screen in the background. Paasonen, 12:56-13:7. See Paasonen, Fig. 10. Thus, when comparing Figures 9 and 10, Paasonen shows that when a pop-up menu is displayed, the rest of the screen, including the currently-running first work process, is dimmed such that the menu is displayed more brightly.

(m) Claim [6]

103. **Claim 6 recites “The method of claim 1, wherein an item of the second work process is displayed relatively brighter or clearer than an item or a screen of a work process other than the second work process.”** This Claim is met by Paasonen.

104. **Paasonen further discloses that an item of the second work process is displayed relatively brighter or clearer than an item or a screen of a**

work process other than the second work process (e.g., “context-specific menu 90” for a browser is displayed brighter than an “application view” for a radio application). See, e.g., Paasonen (emphasis added):

- **Paasonen, Figs. 7–8**

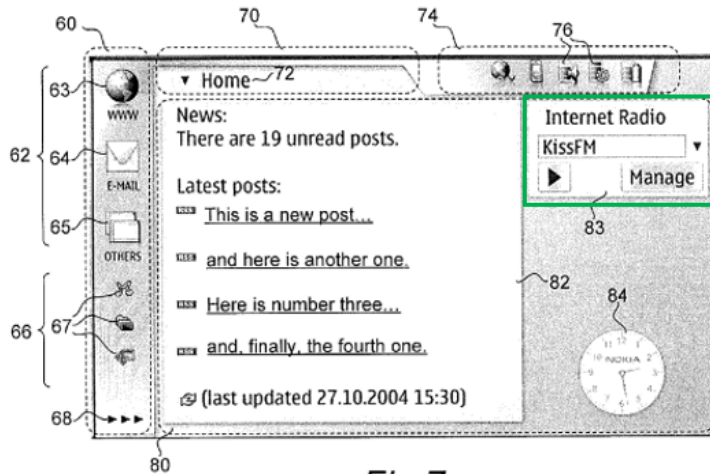


Fig 7

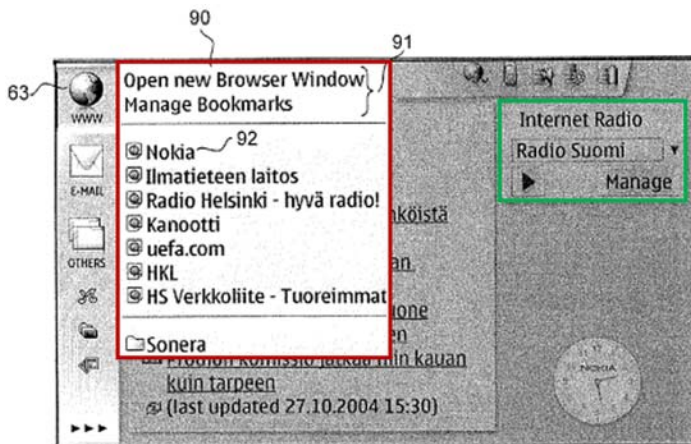


Fig 8

- **Paasonen, 13:16-52** (“...*The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be hierarchical. The lower portion 66 represents an application switcher panel*

with respective icons 67 for each of a plurality of launched applications. ***The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.”***);

-
- **Paasonen, 15:61-16:40** (“During ongoing use of the pocket computer 1, irrespective of whatever other application that is currently active, the **user 9 may always return to the home application by pressing the home key 5d on the front surface 2f of the apparatus housing 2.** Another way of **invoking the home application is through the application switcher menu button 68,** as has been described above.

As seen in FIG. 7, in this embodiment ***the home application contains three application views 82, 83 and 84 on the display 3. Each application view is a downscaled version of the application view of another application 57.*** Thus, among all the functionality nominally provided by such another application 57, ***the application view in the home application will only provide access to limited parts thereof.*** For instance, application view 82 in FIG. 7 represents a news application (e.g. Usenet news) and provides a limited view of this application by displaying the number of unread posts together with a few of the latest posts. Tapping on any of these latest posts will cause presentation of the contents of the post in question. If the user wants to access the complete functionality of the news application, he may switch to this application through e.g. the application switcher menu button 68 (as described above), or the “Others” icon 65 in the upper part 62 of the task navigator 60. In another embodiment, tapping on a post in the application view 82 may directly cause launching (if not already running) of or switching to the news application.

The **application view 83 represents an Internet radio application and gives a limited view of its functionality**. By tapping on a “Manage” button therein, the user may invoke the actual Internet radio application to access its entire functionality. The application view 84 represents a Clock application. The **interaction between such a limited application view 82, 83, 84 and the actual application** it represents may be implemented using push technique, as is readily realized by a skilled person.

In one embodiment, the **user may configure which application views to include** in the home application, and some particulars of them.

Using only limited resources in terms of memory, CPU load and display screen space, the home application gives the user 9 a very convenient overlook view of certain applications that he probably likes to access frequently.”).

105. See my discussion of elements [1.pre]–[1.b]. As explained for [1.pre] above, Paasonen discloses running simultaneous work processes on a mobile terminal. See [1.pre]. As I further explained for elements [1.a]–[1.b], Paasonen discloses displaying a screen of the work processes. See [1.a]–[1.b]. Paasonen further teaches displaying a “context-specific menu 90” of an application, such as the web browser, brighter than an item of another application. Paasonen, Fig. 8, 13:16-52. For example, Figure 8 illustrates the web browser application’s context-specific menu 90 is displayed brighter than the limited “application views,” which would include, e.g., the application view for a media player application. Paasonen, Figs. 7–8, 16:24-29. Thus, the screen as shown in Figure 8 shows both a menu 30 for the browser application, which is an item of the second work process that is shown brighter while the user interacts with that menu, and also shows the home screen “application view” from an audio player, which is the first work process,

such as the internet radio/media player, and further shows that the internet radio application view has been dimmed such that the web browser menu is relatively brighter.

(n) Claim [7]

106. **Claim [7] recites “The method of claim 3, wherein the third command and the fifth command are inputted through at least one of a predetermined key, a user's voice, and a motion of the mobile terminal.”** This Claim is met by Paasonen.

107. **Paasonen further discloses that the third command and the fifth command are inputted through a predetermined key** (*e.g.*, third command inputted through “pressing” “home key 5d” or “center position” key; fifth command inputted using “navigation,” “center position,” or “cancel” key). *See, e.g.*, Paasonen (emphasis added):

- **Paasonen, Fig. 6**

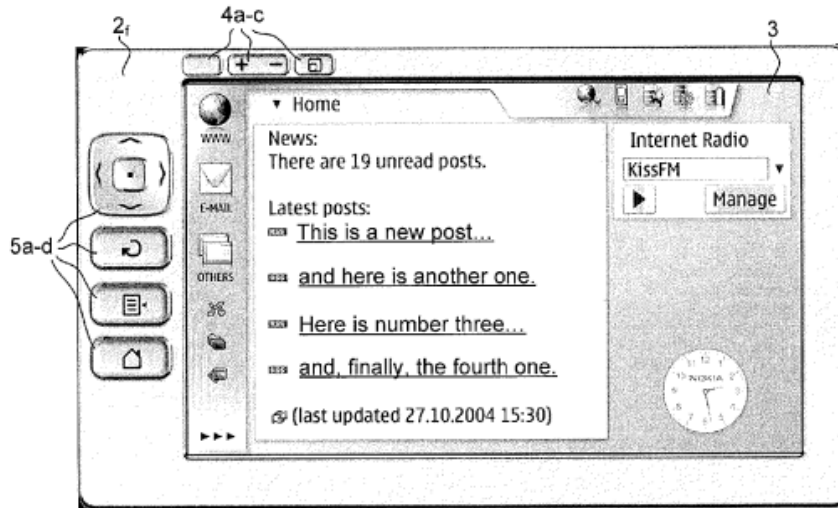


Fig 6

- **Paasonen, 11:2-11** (“Next to the display 3 a plurality of *hardware keys 5a-d* are provided, as well as a speaker 6. More particularly, *key 5a is a five-way navigation key, i.e. a key which is depressible at four different peripheral positions to command navigation in respective orthogonal directions* ("up", "down", "left", "right") among information shown on the display 3, *as well as depressible at a center position to command selection among information shown on the display 3. Key 5b is a cancel key, key 5c is a menu or options key, and key 5d is a home key.*”);
- **Paasonen, 13:16-52** (“...*The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be hierarchical. The lower portion 66 represents an application switcher panel with respective icons 67 for each of a plurality of launched applications. The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the*

associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.”);

- ***Paasonen, 15:61-16:3*** (“[I]rrespective of whatever other application that is currently active, the ***user 9 may always return to the home application by pressing the home key 5d on the front surface 2f of the apparatus housing 2.*** Another way of ***invoking the home application is through the application switcher menu button 68,*** as has been described above.”).
- ***See also Paasonen 11:12-16, 11:12-31, 12:45-55, 14:30-37.***

108. See my discussion of element [1.f] and claim [3]. As explained for [1.f] and claim [3] respectfully, Paasonen discloses a third command for activating a control menu and a fifth command for deactivating the menu. See [1.f], [3]. Paasonen further discloses initiating the user commands using keys such as dedicated, and thus predetermined, “navigation,” “cancel,” and “home” keys. Paasonen, 11:2-11, 13:16-52, 15:61-16:40, Fig. 6. Under Paasonen’s limited “application view” feature discussed for [1.f], a command to activate the application view by returning to the home application is inputted using the “home” key because the “application view” is on the home screen. Paasonen, 15:61-16:3, Fig. 6. To deactivate the application view, the user inputs a command to navigate away from the home application by using a “center position” key to switch to a

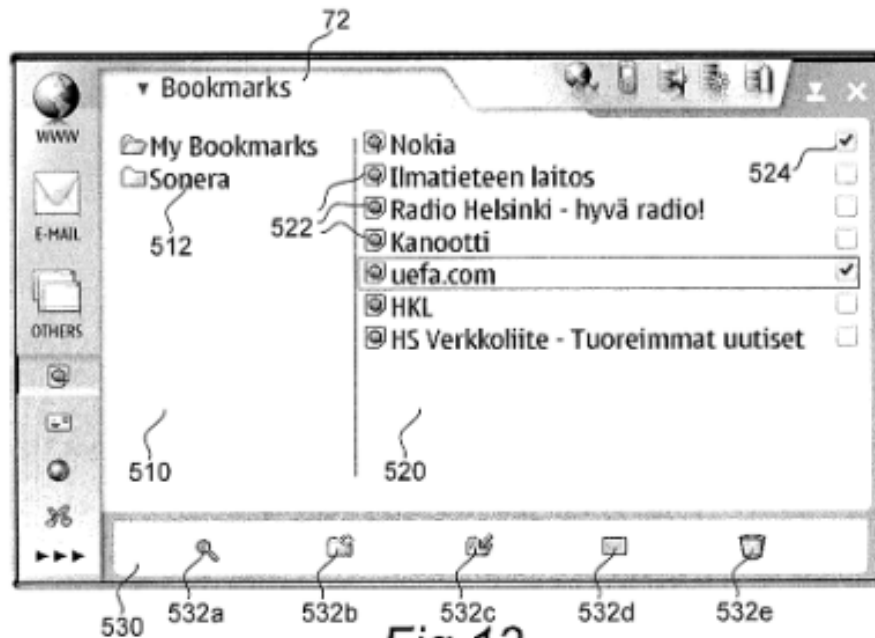
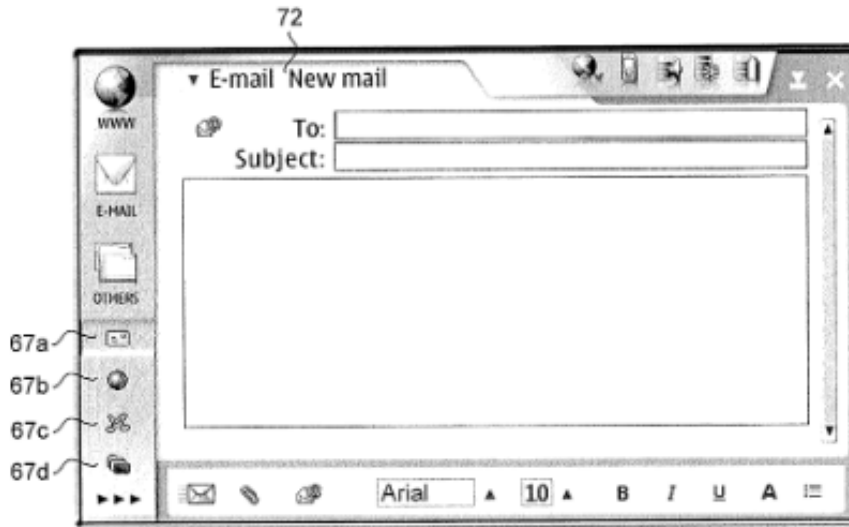
different application. Paasonen, 11:2-11, Fig. 6. And with respect to Paasonen's context-specific menu feature, "command selection" of a context-specific menu item, the selection is inputted using a "center position" key. Paasonen, 11:2-11, 13:16-52, 15:61-16:23. To deactivate the context-specific menu, a user presses a "cancel" key to exit the menu or navigates away from the menu by select switch to a different application. Paasonen, 11:2-11.

(o) Claim [8]

109. **Claim [8] recites "The method of claim 1, wherein, in response to a repetition of the first command, performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal." This is met by Paasonen.**

110. **Paasonen further discloses in response to a repetition of the first command (see [1.c]; e.g., reopening "bookmark manager," after four other applications opened), performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal (see [1.c]; icon for "new application" is "insert[ed]"; oldest icon is "shifted out"). See, e.g., Paasonen (emphasis added):**

- Paasonen, Figs. 12-13



- Paasonen, 9:5-16 (“The graphical user interface may be further adapted, upon launching of a new application, to insert an icon associated with said new application at said first position in the application switcher panel while shifting the positions of existing icons in the application switcher panel by one position backwards.”)

In one embodiment, ***only a predetermined maximum number of positions for icons may be allowed*** in said application switcher panel wherein, for ***an icon that has been shifted out from the application switcher panel***, the application associated therewith may be activated through selection of a menu item in a menu on said display.”);

- **Paasonen, 13:16-52** (“...***The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91***, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be hierarchical. The lower portion 66 represents an application switcher panel with respective icons 67 for each of a plurality of launched applications. ***The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.***”);
- **Paasonen, 14:38-48** (“[I]n the disclosed embodiment a maximum of ***four different running applications will be represented by respective icons 67 in the application switcher panel 66***. The order among the icons 67 is such that ***the icon for the most recently active application will be shown at the topmost position...***”).
- **See also Paasonen Figs. 7-9, 11, 14:60-15:18.**

111. See my discussions of elements [1.a] and [1.c]. As explained for [1.a]

above, Paasonen discloses displaying icons corresponding to different work

processes. *See* [1.a]. As I further explain for [1.c] above, Paasonen also discloses performing sequential display of icons in response to a first command. *See* [1.c]. Paasonen further discloses that the application icons disappear and appear on the application switcher panel. Paasonen, 14:38-48. Paasonen's application switcher panel shows "a maximum of four" icons corresponding to "running applications," leading with the most-recently active application. Paasonen, 14:38-48. Thus, when a user repeats the first command by re-opening the bookmark manager application, for example, from the screen in Figure 12 in which the bookmark manager application icon is not shown, the fourth most recently active application icon in position 67d is "shifted out of the application panel." Paasonen, 9:5-16, Figs. 12-13. The bookmark manager application icon then appears in the "topmost" position of the panel and the application icon in the fourth position disappears. Paasonen, 14:38-48.

(p) Claim [9]

112. **Claim [9] recites "The method of claim 1, wherein an icon to be displayed in the designated portion is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command."** This is met by Paasonen.

113. **Paasonen discloses wherein an icon to be displayed in the designated portion** (*see* [1.c]); the icon for the second work process is displayed in

the “second position” (67b) prior to “shift[ing]” icons downward) **is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command** (see [1.c]; shifting downwards causes a different icon to replace the icon in “second position” (67b)). *See, e.g., Paasonen* (emphasis added):

- **Paasonen, Figs. 12–13**

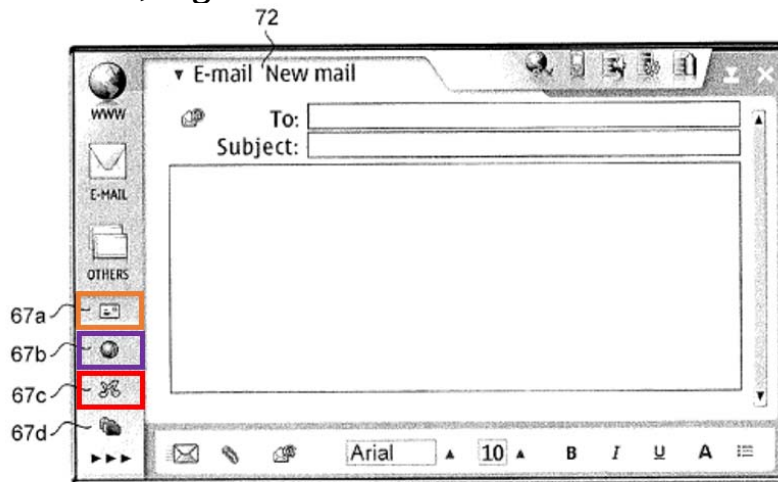


Fig 12

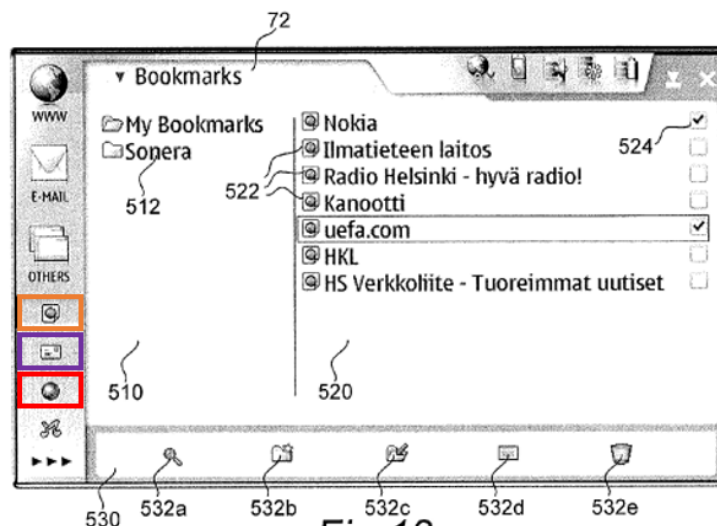


Fig 13

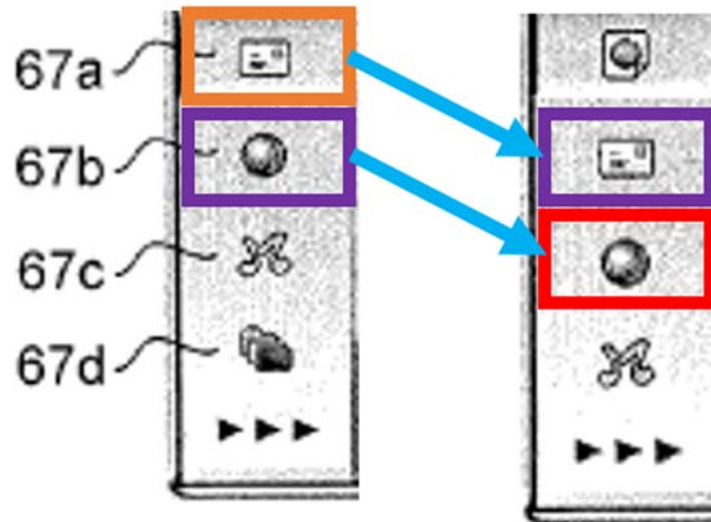
- **Paasonen, 9:5-16** (“The graphical user interface may be further adapted, *upon launching of a new application*, to *insert an icon associated with said new application at said first position in the application switcher panel while shifting the positions of existing icons in the application switcher panel by one position backwards*.
In one embodiment, *only a predetermined maximum number of positions for icons may be allowed* in said application switcher panel wherein, for *an icon that has been shifted out from the application switcher panel*, the application associated therewith may be activated through selection of a menu item in a menu on said display.”);
- **Paasonen, 13:16-52** (“...*The upper portion 62 contains icons 63-65 which when selected will open a task-oriented, context-specific menu 90 to the right of the selected icon (see FIG. 8, FIG. 11). The context-specific menu 90 will contain a plurality of task-oriented menu items 91*, and the user may navigate among these menu items and select a desired one either by the navigation key 5 a or by pointing at the display 3. The menu 90 may be hierarchical. The lower portion 66 represents an application switcher panel with respective icons 67 for each of a plurality of launched applications. *The topmost icon 63 is used for accessing tasks related to information browsing. The available tasks are presented as menu items 91 in menu 90, as seen in FIG. 8. More particularly, the user 9 may choose between opening a new browser window (FIG. 9), or managing bookmarks. Selecting of any of these menu items 91 will cause launching of the associated application (a browser application as seen in FIG. 9 or a bookmark manager as seen in FIGS. 13-14), or switching to such application if it is already included among the active ones, and also invocation of the appropriate functionality therein. In addition, the menu 90 contains a set of direct links 92 to certain web pages. In the disclosed embodiment, this set includes bookmarks previously defined by the user 9, but in other embodiments it may include the most recently visited web sites. The second icon 64 is used for accessing tasks related to electronic messaging, as is seen in FIGS. 11 and 12.*”);
- **Paasonen, 15:6-18** (“Now, *when the user 9 invokes the messaging application by selecting the menu item 93 in the afore-described task-oriented menu 90, the messaging application becomes active and its icon*

takes the topmost position 67 a, as seen in FIG. 12. At the same time, the existing icons 67 a-c of FIG. 11 are shifted one vertical position downwards, so that the web browser icon (formerly at 67 a) takes the second position at 67 b, the audio player icon moves to the third position 67 c, and the file manager icon goes to the lowermost position 67 d. The formerly shown image viewer icon disappears from the application switcher panel 66, but the image viewer application is still running.”).

- ***See also Paasonen 14:60-15:5.***

114. See my discussion of element [1.c]. As explained for [1.c] above, Paasonen discloses sequential display of icons in the designated portion in response to a first command. See [1.c]. Paasonen discloses running several applications, or work processes, on its terminal besides the web browser and music player applications. Paasonen, 11:48-65 (“[E]xamples of applications are an Internet/WWW/WAP browser application, a contacts application, a messaging application (email, SMS, MMS), a calendar application,” “a music player application, a media streaming application,” while another application is “active.”). These applications are represented as icons in Paasonen’s application switcher panel if they are one of the most-recently active applications and in the order based on how recently active the application is. Paasonen, 14:38-48. Thus, initiating a first command to select and launch an application from the screen illustrated in Figure 12—such as the “bookmark manager” application—causes the icon for the web browser application (i.e., the second work process) to shift from “**second**

position” 67b to **“third position”** 67c (i.e., the designated portion) . Paasonen, 11:48-65, 15:6-18, Figures 12-13.



At the same time, the “messaging application” icon (i.e., an icon different from the icon corresponding to the second work process) that was located in the **“first position”** 67a will shift down into **“second position”** 67b and replace the icon that used to be there. Paasonen, 9:5-16, 15:6-18.

(q) Claim [10]

115. **Claim [10] recites “The method of claim 1, wherein the displayed icon is displayed together with at least a part of a screen image of a work process in the display.” This is met by Paasonen.**

116. **Paasonen further discloses that the displayed icon (see [1.a], [1.c]) is displayed together with at least a part of a screen image of a work process**

in the display (e.g., “application switcher panel” is displayed with a “web browser”). See, e.g., Paasonen (emphasis added):

- **Paasonen, Fig. 9**

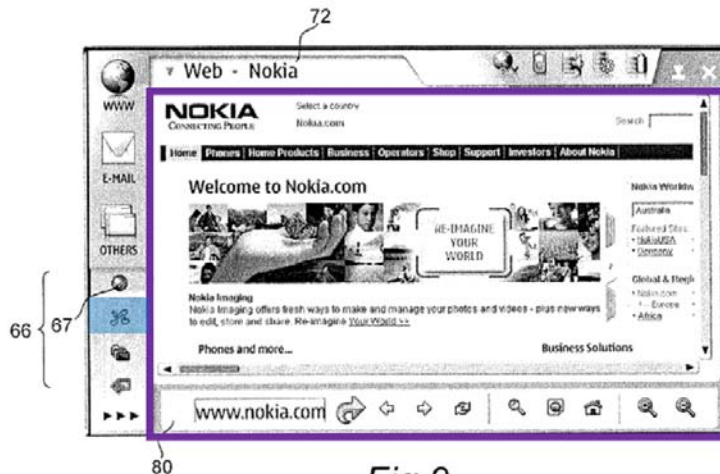


Fig 9

- **Paasonen, 12:56-13:7** (“With reference to FIG. 7, *the display screen layout of the display 3 is divided into four main areas: a task navigator 60, a title area 70, a status indicator area 74 and an application area 80.* The *application area 80 is used by a currently active application to present whatever information (content) is relevant* and also to provide user interface controls such as click buttons, scrollable list, check boxes, radio buttons, hyper links, etc, which allow the user to interact with the currently active application by way of the stylus 9c. *One example of how a currently active application, in the form of a web browser, uses the application area 80 in this manner is shown in FIG. 9.... In addition, as seen in FIG. 10, by tapping in the title area 70, the user may access an application menu 73 of the currently active application.*”);
- **Paasonen, 14:22-37** (“*The lower portion 66 of the task navigator 60 will now be described in more detail. As already mentioned, it represents an application switcher panel with respective icons 67 for each of a plurality of launched applications, i.e. running applications that are executed by the controller 50. Among such running applications, one will be active in the sense that it has control over the application area 80 on the display 3.*”);
- **15:6-18** (“*[W]hen the user 9 invokes the messaging application by selecting the menu item 93 in the afore-described task-oriented menu 90,*

the messaging application becomes active and its icon takes the topmost position 67a, as seen in FIG. 12. At the same time, the existing icons 67a-c of FIG. 11 are shifted one vertical position downwards, so that the web browser icon (formerly at 67a) takes the second position at 67b, the audio player icon moves to the third position 67c, and the file manager icon goes to the lowermost position 67d....”).

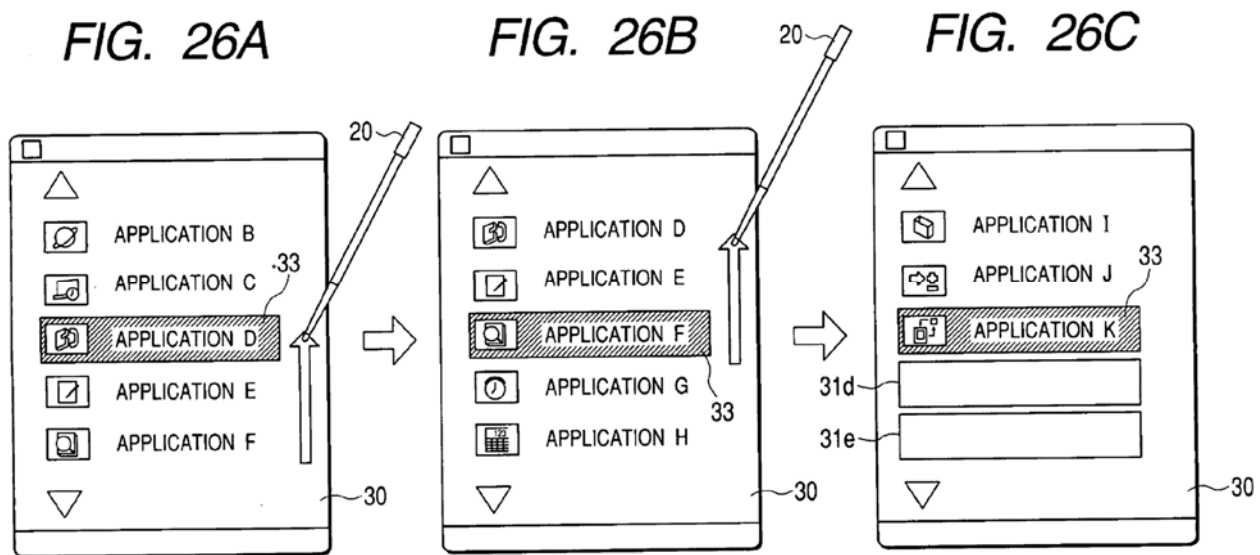
117. See my discussion of elements [1.a] and [1.c]. As explained for element [1.a] and [1.c] above, Paasonen discloses displaying icons corresponding to work processes, including the “audio player” and “browser” icons. See [1.a], [1.c]. As I further explained for element [1.c] above, Paasonen discloses displaying a screen image of a work process. See [1.c]. Paasonen further discloses the terminal’s “display screen layout” is divided into a “task navigator 60,” which includes the “application switcher panel,” “title area 70,” “status indicator area 74,” and “an application area 80.” Paasonen, 12:56-13:7. The “application switcher panel” including the application icons, is displayed beside “application area 80.” Paasonen, Fig. 9, 12:56-13:7, 14:22-29. As shown in Figure 9, the “audio player” and “browser” icons are displayed beside an activated “web browser” application screen in application area 80. Paasonen, Fig. 9, 14:60-15:5, 15:6-18.

B. Ground 2: Paasonen in view of Nakano Renders Obvious Claims 1-10

1. Overview of Nakano and Motivation to Modify Paasonen with Nakano's Teachings

118. As discussed in §IX.A for limitation [1.c] and claims 8-9, Paasonen discloses sequentially displaying icons in the designated portion of the display. To the extent further disclosure of this element is required, modifying Paasonen with Nakano renders obvious these elements. Nakano is titled “Portable information terminal, display control device, display control method, and computer readable program therefor” and teaches a “menu display” which provides a user of a “portable information terminal” a sequential “arrangement of menu items” with icons. Nakano, Figs. 26A-26C, [0005], [0016], [0046], [0288], [0301], [0303]. Nakano teaches presenting a user with a list of icons on a “menu display” that serves as an “easy-to-use” list of applications available on the portable terminal. Nakano [0005], [0016]. Each icon “represent[s an] application software program” available on the terminal. Nakano [0005]. The icons are displayed in an application list, shown in Figure 26, that a user easily “scroll[s]” through by performing a “tap-in” and “drag” operation with a finger or stylus. Nakano [0046]. As “a result of scroll[ing]” through the application list, the icons are cycled vertically through each position in the application list. See Nakano, Figs. 26A-26C. As exemplified in the sequence in Figures 26A-26C, the “menu display” is

updated continuously as the top icons, e.g., applications “B” and “C” in Figure 26A, disappear from the list and new bottom icons, e.g., applications “I,” “J,” and “K” in Figure 26C appear:



Nakano, Figs. 26A-26C, [0046]. Nakano permits users access to an “easy understanding . . . of menu items” representing the available applications that can be selected by the user on the device. Nakano [0016].

119. Paasonen and Nakano are analogous to and in the same field as the ’631 patent, which includes mobile interfaces (*e.g.*, user interfaces on mobile devices), and are reasonably pertinent to the alleged problems identified in the ’631 patent—*e.g.*, providing interfaces for multi-tasking mobile terminals. *See, e.g.*, ’631 Patent, 2:32-51 (“In the conventional method for selecting the work process, when selecting a work process that is concurrently performed, the user has to select a new work process through a displayed menu window. Also, in order to terminate

a predetermined work process among a plurality of work processes that are concurrently performed and select a new work process, the user has to select the new work process through the menu window.”); Paasonen, 1:6-9 (“The present invention generally relates to portable electronic equipment, and more particularly to a pocket computer having a graphical user interface. The invention also relates to various methods of operating the user interface.”), 3:41-4:25 (“This same basic need is even more obvious in all handheld devices that have windowed graphical user interfaces and limited available screen space . . . [¶] A problem with the prior art . . . is how to efficiently and intuitively switch between running applications on a hand-held device such as a pocket computer.”); Nakano [0001] (“[P]ortable information terminal . . . and a display control method for allowing a computer to perform display control of menu display.”), [0006] (“The size of a physical display screen . . . is limited, for example, within the palm of a hand of an adult in order to focus on the portability.”), [0014]-[0015] (“[M]enu items which extend off screen must be displayed by scrolling through the screen . . . [¶] [T]he display screen is scaled up or down through zoom-in or zoom-out operation in the related art examples For a device with a small display size such as a PDA, simply arranging menu items on a plane provides poor visibility.”).

120. Paasonen teaches that the four most-recently active applications are displayed sequentially in the “application switcher panel.” Paasonen, 14:22-29.

Nakano discloses that a user may scroll through an application menu and view applications that may not have been initially displayed on the screen. Paasonen's "application switcher menu button 68" permits users to view running applications not displayed in its switcher panel in a "popup window." Paasonen, 15:19-30. However, a "popup window" obscures the "application area 80" where the "currently active application" content is presented. Paasonen, Fig. 7., 12:60-65, 15:19-30. Modifying Paasonen's switcher panel with Nakano's scrolling teachings advantageously permits users to browse more applications without obscuring the application area with a popup menu. A person of ordinary skill in the art would have been motivated to modify Paasonen's application switcher panel with Nakano's scrolling through a menu of application icons teachings in order to improve the user's access to recently-active applications by enabling the user to scroll through all available applications rather than limiting the menu to the four most recently used applications. See Nakano [0015]-[0016]; Paasonen, 14:38-48 ("[I]n the disclosed embodiment a maximum of four different running applications will be represented by respective icons 67 in the application switcher panel 66."). Modifying Paasonen with Nakano's sequential display teachings addresses the challenge of "arrang[ing] a large number of menu items" and advantageously "enhance[s] the operability" of the device. Nakano [0015]. Indeed, Nakano, like Paasonen, teaches that a finite number of icons are displayed in an area, such that it

would have been a straightforward modification of Paasonen’s application switcher panel to allow that panel to scroll, obtaining the benefits of Nakano’s teachings while still maintaining the visibility of the currently-running application. Nakano [0046]. Indeed, Nakano discloses that its teachings “may be applied to other electronic devices and information terminal devices which provide[] menu displays”—affirmatively providing a further motivation to modify a device such as Paasonen’s with these teachings. Nakano [0148].

121. A person of ordinary skill in the art also would have had a reasonable expectation of success when modifying Paasonen’s application switcher with Nakano’s scrolling teachings as scrolling was a well-known and well-understood technique for navigating items on a virtual display’s menu. *See, e.g.*, Tu (Ex. 1008), [0031] (“The medium icons 312-324 can be moved or scrolled across the menu screen 302.”); Tornqvist (Ex. 1017), 3:24-26 (“The present invention relates to a method and an arrangement for scrollable cross point navigation on a user interface in order to select a feature comprised in a combination of two object fields.”); Zadesky (Ex. 1018), [0005] (“[T]he present invention relates to touch pads having scrolling features.”).

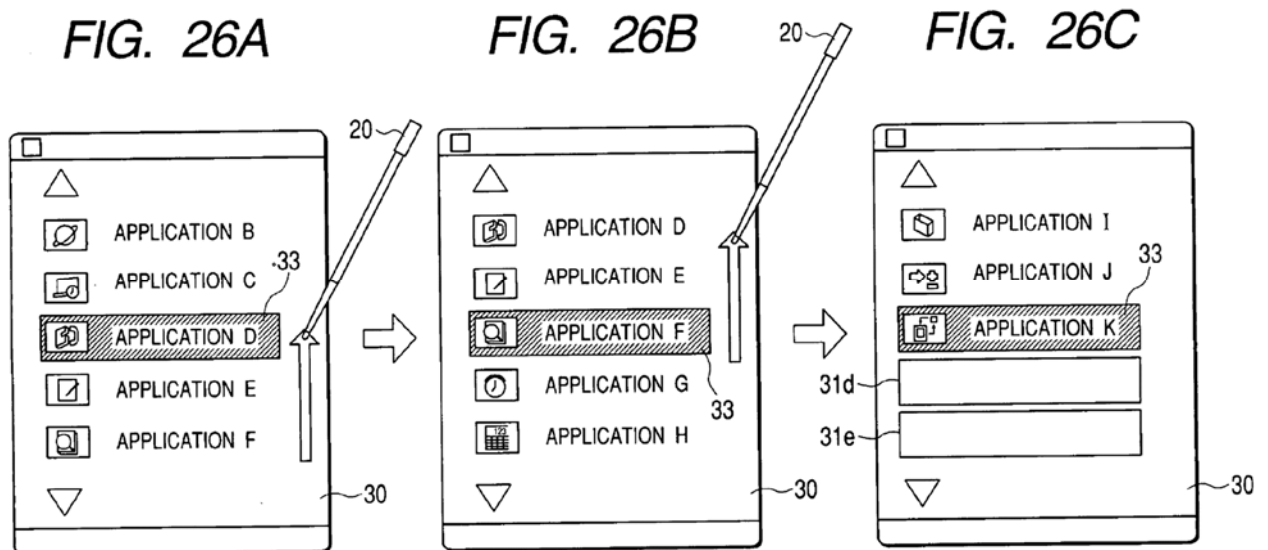
2. Invalidity of Claims 1-10 Over Paasonen in view of Nakano

(a) Element [1.c]

122. Element [1.c] recites “in response to a first command for sequentially displaying icons, performing sequential display of icons such that an icon corresponding to the second work process is displayed, wherein the icon corresponding to the second work process is displayed in the designated portion of the display.” This element is met by Paasonen in view of Nakano.

123. In addition to the teachings from Paasonen that I discussed above in Section IX.A.2(d), **Nakano discloses in response to a first command for sequentially displaying icons, performing sequential display of icons** (e.g., scrolling through application icons in response to a “scroll operation instruction”). See, e.g., Nakano (emphasis added):

- Nakano, Figs. 26A–26C



- **Nakano, [0005]** (“In the related art menu display by the launcher, icons representing application software programs and/or documents and their names are displayed in sets (the sets are hereinafter referred to as ‘menu items’), arranged like tiles on a plane. The user selects any menu item by way of operation of a button such as a lever and across-hair key, or selection using a pointing device such as a stylus pen or a tracking pad.”);
- **Nakano, [0016]** (“The invention has been accomplished in order to solve the aforementioned problems and aims at providing easy-to-use menu display which allows easy understanding of arrangement of menu items used for a portable information terminal.”);
- **Nakano, [0046]** (“The drag operation refers to operation of drawing a line on a touch panel with a stylus pen or a finger of the user. In the drag operation, a moment a stylus pen comes into contact with a touch panel is referred to as a ‘tap-in’ and a moment a stylus pen is released from a touch panel after moving the stylus pen with the stylus pen kept in contact with the touch panel is referred to as a ‘tap-out.’ The operation of causing a stylus pen to come into contact with a touch panel instantaneously and release the stylus pen is called a ‘tap.’ According to this invention, a drag operation is available across a predetermined display area where a plurality of information display entities are displayed. Scroll display of information display entities is executed by a drag operation. Thus, the user has only to perform a drag operation like drawing an arbitrary line in order to perform scroll operation. The display screen of a personal information terminal device is small so that the predetermined display area maybe set to the entire display screen of the display. For example, a menu display entity may be used as an information display entity and in case a menu screen is to be displayed, the entire display screen may be used to display the menu screen, and a scroll operation (or a drag operation) is allowed across the screen. Further, as a result of scroll, an information display entity (menu display entity) displayed in the position of a candidate for selection is selected as a candidate for selection. Thus, a single drag operation serves as a scroll operation instruction and selection of an information display entity (menu display entity).”);
- **Nakano, [0288]** (“A menu display area 31 c is also used as a ‘selection candidate display area 32’. In the selection candidate display area 32, a focus 33 (cross-hatching) is shown. The menu display entity displayed in the area is a ‘candidate for selection’.”);

- **Nakano, [0301]** (“That is, as shown in FIG. 27A, dragging in the downward direction scrolls through the menu display entities in the downward direction. As shown in FIG. 27B, when the uppermost menu display entity (Application A in the figure) reaches the selection candidate display area 32, that is, the menu display area 31c, the scroll comes to a halt.”);
- **Nakano, [0303]** (“FIG. 28 illustrates the concept of scrolling through menu display entities.”).

124. In addition to Paasonen’s teachings, as discussed in Section IX.A.2(d), Nakano discloses a “scroll operation instruction” that scrolls through the “menu display” icons representing applications. Nakano [0005], [0046]. As shown by the sequence in Figures 26A-26C, scrolling through the menu display sequentially displays the icons. Nakano, Figs. 26A-26C. For example, Applications “D,” “F,” and “K” are cycled through the “candidate display area” in the middle of the menu screen shown in the sequence. Nakano, Figs. 26A-26C, [0288]. The user initiates the scroll instruction with a “tap” or “drag operation.” Nakano [0046]. As discussed in Section IX.B.1, a person of ordinary skill in the art would have been motivated to modify Paasonen’s application switcher with Nakano’s scrolling teachings in order to permit users to easily and intuitively browse through application icons that are not readily displayed in the application switcher panel. Paasonen’s switcher panel displays a “maximum of four” application icons, but Nakano’s scrolling teachings would permit access to more icons. See Paasonen, 14:38-48. Scrolling was a common and well-understood method to permit users to navigate items on a screen, thus a person of ordinary

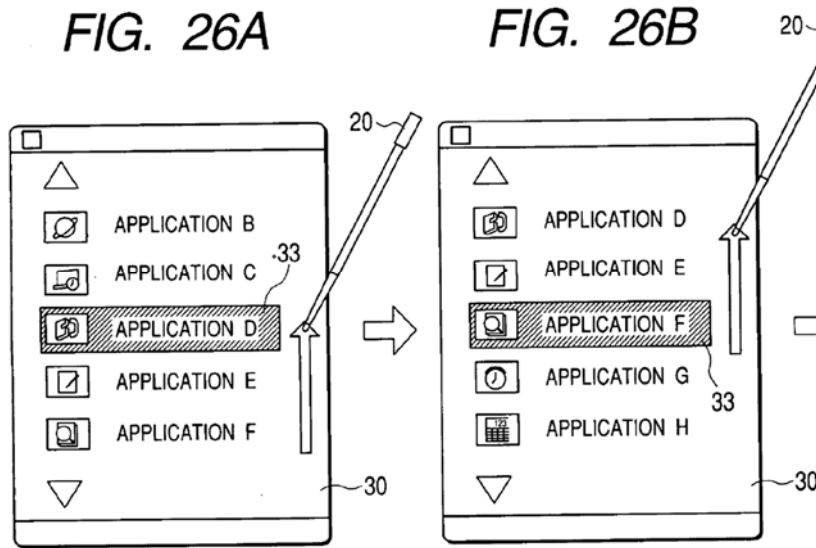
skill in the art would have reasonably expected success when modifying Paasonen's switcher panel with Nakano's teachings.

(b) Claim [8]

125. **Claim [8] recites “The method of claim 1, wherein, in response to a repetition of the first command, performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal.”** This element is met by Paasonen in view of Nakano.

126. In addition to the teachings from Paasonen that I discussed above in Section IX.A.2(o), **Nakano discloses in response to a repetition of the first command** (*e.g.*, additional “scrolling” command), **performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal** (*e.g.*, “scroll[ing] through the menu display entities” causes some icons to disappear as others appear in Fig. 26B). *See, e.g.*, Nakano (emphasis added):

• **Nakano, Figs. 26A–26B**



- **Nakano, [0157]** (“The second method is an input with the touch panel 104 and the stylus pen 106. In the display screen are displayed a scroll-up icon 136 and a scroll-down icon 138. When the user touches the touch panel 106 with the stylus pen 106 with an intention to touch an icon, menu items in the selection state are deselected. Then, in accordance with the number of time the lever is tilted or duration the lever is tilted, the user can place the menu under moving display in the upward or downward direction (upward/downward direction on the display screen). The user then taps (patting on the touch panel 104 with the stylus pen 106).”);
- **Nakano, [0184]** (“To be more specific, for example, in case the selecting operation is a short titling (click) of the side controller 180 in the upward/downward direction or tapping on the scroll-up icon 136 or down-scroll icon 138, a menu item positioned in the upward/downward direction the number of operations away from the menu item currently selected is specified.”);
- **Nakano, [0229]** (“To be more specific, in the case of an input from the side controller 108, or scroll-up icon 136 or scroll-down icon 138 (YES in step S104), the menu item selector 22 changes the specified menu number 532 in accordance with the input duration (step S110). Except for a hold-down input (NO in step S106), the menu item selector 22 increments the specified menu number 532 by “1” in the input direction in accordance with the specification (step S108).”)

- **Nakano, [0292]** (“In the menu screen 30, dragging the stylus pen 20 in the upward direction scrolls through the menu display entities in the menu screen 30 as shown in FIG. 26B.”).

127. In addition to Paasonen’s teachings, as discussed in Section IX.A.2(o), Nakano discloses the scroll operation causes menu display icons to disappear from the top of the menu screen and appear at the bottom of the menu screen. Nakano, Figs. 26A-26C, [0229]. Each scroll-down or scroll-up operation “increments” menu icon positions by 1, such that, depending on the direction, an application icon disappears from the screen while another appears on the screen. Nakano, [0157], [0184], [0229]. As discussed in Section IX.B.1, a person of ordinary skill in the art would have been motivated to modify Paasonen’s application switcher panel teachings with Nakano’s scrolling teachings in in order to make navigating application icons on Paasonen’s display easier and make it possible to access more applications through the switcher panel.

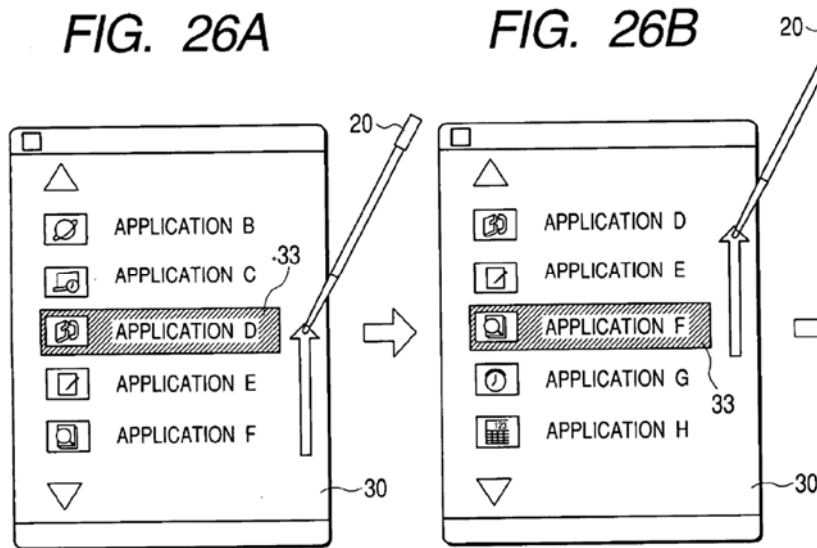
(c) Claim [9]

128. **Claim [9] recites “The method of claim 1, wherein an icon to be displayed in the designated portion is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command.”** This Claim is met by Paasonen in view of Nakano.

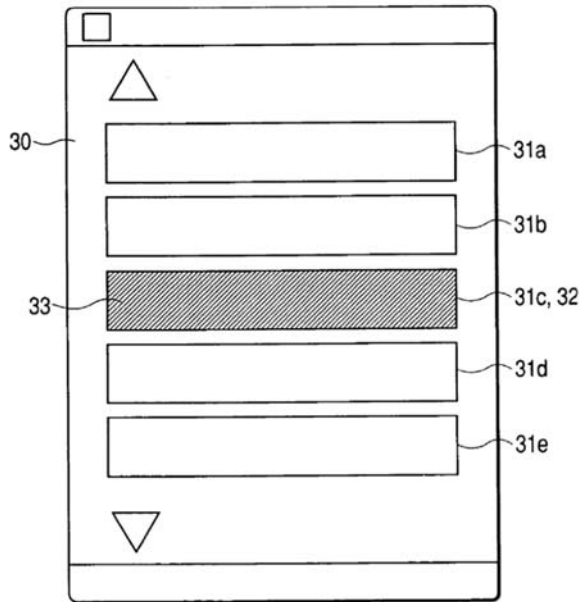
129. In addition to the teachings from Paasonen that I discussed above in Section IX.A.2(p), **Nakano discloses wherein an icon to be displayed in the**

designated portion is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command (e.g., “scroll[ing] through the menu display entities” causes different application icons to be displayed in each “menu display area”). See, e.g., Nakano (emphasis added):

- Nakano, Figs. 26A-26B



- **Nakano, Fig. 25**
FIG. 25



- **Nakano, [0292]** (“In the menu screen 30, dragging the stylus pen 20 in the upward direction scrolls through the menu display entities in the menu screen 30 as shown in FIG. 26B.”);
- **Nakano, [0293]** (“In FIG. 26B, the menu display area 31 a displays “Application D”, the menu display area 31 b displays “Application E”, the menu display area 31 c displays “Application F”, the menu display area 31 d displays “Application G”, and the menu display area 31 e displays “Application H”, respectively.”);
- **Nakano, [0297]-[0298]** (“In FIG. 26B, further dragging in the upward direction further scrolls through the menu entities on the screen in the upward direction. As shown in FIG. 26C, when the lowermost menu display entity (Application K in the figure) reaches the selection candidate display area 32, the scroll comes to a halt. In FIG. 26C, the menu display area 31 a displays “Application I”, the menu display area 31 b displays “Application J”, and the menu display area 31 c displays “Application K”, respectively.”).

130. In addition to Paasonen’s teachings, as discussed in Section IX.A.2(p), Nakano discloses replacing menu display icons in response to a scroll

operation instruction. Nakano [0297]-[0298]. As shown in the scroll-up sequence shown in Figures 26A and 26B, “Application E” replaces “Application D” and “Application F” replaces “Application E.” Nakano, Figs. 26A-26C, [0293]. As discussed in Section IX.A.2(p), a person of ordinary skill in the art would have been motivated to modify Paasonen with Nakano’s scrolling teachings.

C. Grounds 3–4: Paasonen in view of Aaltonen (Ground 3) and further in view of Nakano (Ground 4) Render Obvious Claims 4 and 7

1. Overview of Aaltonen and Motivation to Modify Paasonen and Nakano with Aaltonen’s Teachings

131. As discussed in §§IX.A–B, Paasonen (and Nakano) discloses and/or renders obvious an MP3 play process (claim 4) and using a key, user’s voice, or motion to input commands to a mobile terminal (claim 7). To the extent further disclosure of these elements is required, modifying Paasonen (and Nakano) with Aaltonen’s MP3 player and input means teachings further renders obvious these claims. Aaltonen is titled “Stripe user interface” and discloses a “user interface” for a “portable terminal device” that “provides the availability of important information” on a device to the “display.” Aaltonen, Fig. 22G, [0053], [0050], [0054], [0060], [0121], [0158]. Aaltonen further teaches the “user interface” can be implemented to function on a device with a plurality of applications including a “mobile telephone” and an “MP3” player. Aaltonen [0053], [0121]. The “user

interface” allows for the “simultaneous interaction” of different applications “available through a device.” Aaltonen [0050]. Aaltonen teaches that for a user to interact with the “user interface,” there are several “input means” for control including both “a touch sensitive display” and a “voice command ability.” Aaltonen [0053].

132. Paasonen and Aaltonen are analogous to and in the same field as the ’631 patent, which includes mobile interfaces (*e.g.*, user interfaces on mobile devices), and reasonably pertinent to the ’631 patent’s alleged problems. *See, e.g.*, ’631 Patent, 2:32-51 (“In the conventional method for selecting the work process, when selecting a work process that is concurrently performed, the user has to select a new work process through a displayed menu window. Also, in order to terminate a predetermined work process among a plurality of work processes that are concurrently performed and select a new work process, the user has to select the new work process through the menu window.”); Paasonen 1:6-9 (“The present invention generally relates to portable electronic equipment, and more particularly to a pocket computer having a graphical user interface. The invention also relates to various methods of operating the user interface.”), 3:41-4:25 (“This same basic need is even more obvious in all handheld devices that have windowed graphical user interfaces and limited available screen space . . . [¶] A problem with the prior art . . . is how to efficiently and intuitively switch between running applications on

a hand-held device such as a pocket computer.”); Aaltonen [0003] (“[T]he disclosed embodiments are related to a user interface for a mobile computing device.”), [0007] (“[I]f an event requires more screen ‘real estate’ for displaying the information associated with the event, a pop-up window is used. However, pop-up windows tend to block the underlying view and user interface objects under the pop-up are not visible. . .”).

133. Paasonen discloses a “music player application,” but does not explicitly identify any music file format. See Paasonen 11:48-65. A person of ordinary skill in the art would have been motivated to modify Paasonen’s music player with Aaltonen’s teachings of “MP3” playback functionality in order to enable playback of MP3 files. Aaltonen [0121]. A person of ordinary skill in the art would have had a reasonable expectation of success when modifying Paasonen with Aaltonen’s MP3 teachings because MP3 files were a well-known music file format and conventional in the art. *E.g.*, Seo (Ex. 1019), [0005] (“[P]rocessing a multitasking program, e.g., downloading a file while playing a moving picture expert group layer 3 (MP3) and editing the Short Message Service (SMS) or Multimedia Message Service (MMS) simultaneously” in a “next generation mobile telecommunication terminal.”); Marriott, [0071] (“[M]edia player 100 is a pocket sized hand held MP3 music player that allows a user to store a large collection of music”).

134. A person of ordinary skill in the art would have been further motivated to modify Paasonen's user input teachings with Aaltonen's teachings of using "voice" commands as a method for inputting commands to the user interface, including commands for activating and deactivating control menus. See Aaltonen [0053]. Accepting user input through voice commands was a known feature for its intended purpose and advantageously permits hands-free operation of mobile devices in a convenient way. See Aaltonen [0121]. The use of voice commands on mobile devices was well-understood in the art. *E.g.*, Hon (Ex. 1020), [0004] (describing "[u]sing voice commands" to "navigate the web" on a mobile phone); Viitamaki (Ex. 1021), [0087] (accepting "user input...voice command" to activate mobile terminal). A person of ordinary skill in the art therefore would have reasonably expected success when modifying Paasonen's application control menus with Aaltonen's voice command teachings.

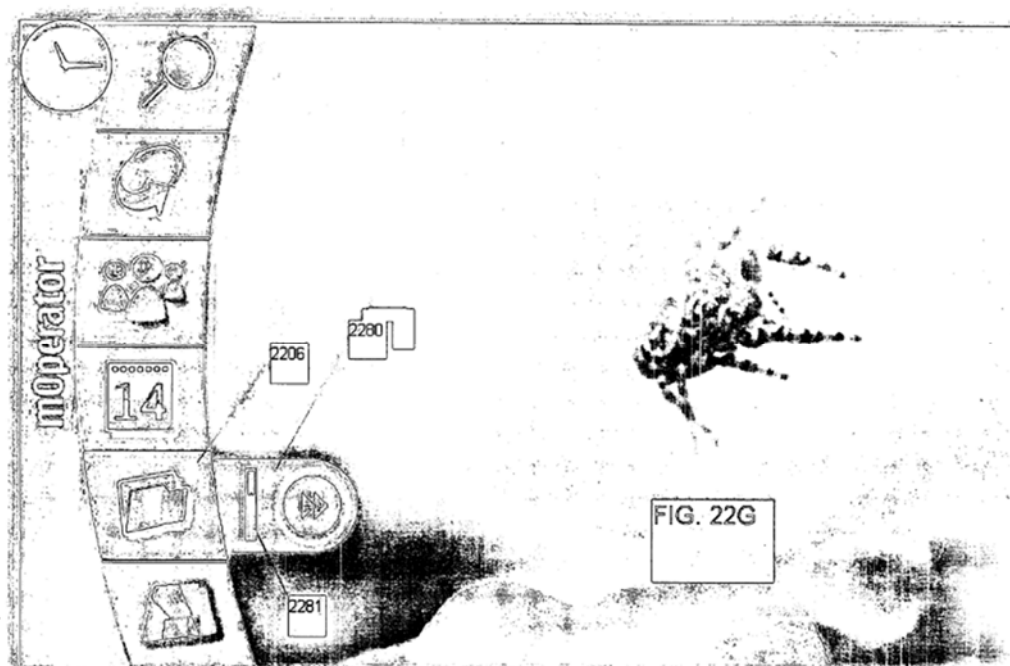
2. Invalidity of Claims 4 and 7 over Paasonen in view of Aaltonen (Ground 3) and further in view of Nakano (Ground 4)

(a) Claim [4]

135. **Claim [4] recites "The method of claim 1, wherein the first work process is an MP3 play process."** This element is met by Paasonen (or Paasonen in view of Nakano) in further view of Aaltonen.

136. In addition to the teachings from Paasonen that I discussed above in Section IX.A.2(k), **Aaltonen discloses an MP3 play process** (e.g., “an ‘MP3’ playback” “action”). See, e.g., Aaltonen (emphasis added):

- **Aaltonen, Fig. 22G**



- **Aaltonen, [0050]** (“The user interface of the present invention generally provides the availability of important information in a device in a single glance. The disclosed embodiments provide for simultaneous interaction between categories of information stored on or available through a device, which can include information related to content, users, tasks, environment and other system information and applications.”);

- **Aaltonen, [0053]** (“The user interface of the present invention is generally provided on a display of an electronic device, such as for example, a portable terminal device, including a mobile telephone. FIG. 1A illustrates a schematic diagram of one embodiment of an electronic device 100 incorporating features of the present invention. The device 100 can be a wireless terminal device operating in a communication system such as for example, a GSM, GPRS, UMTS or Bluetooth system. The electronic device can also be for example a hand-held, portable or desktop computer or a game device or console, or a personal digital assistant (“PDA”). In alternate embodiments, the user interface of the present invention can be implemented on any content and task driven electronic device. The device 100 generally comprises a processor 101 and a memory 102 for operating tasks of the device 100 and for running applications 103 stored in the device. For operating in a communications system, the device can include a transceiver 104 and an antenna 105. For data input, the device 100 can include a keyboard 106, which can be a soft key keyboard or touch sensitive area on a surface of the device providing visual symbols or other indicators for the user to select by touching the symbol. The device 100 can also include other input means, such as for example a joystick, a rocker switch controller, a touch sensitive display or voice command ability. For data output, the device 100 includes a display 107 that can be a monochrome or color display, an LCD panel, a touch sensitive panel or other suitable display as well as vibration motors for tactile output. In one embodiment, the processor 101 can accommodate other input/output devices 109 as well as a speaker/microphone 108.”);
- **Aaltonen, [0054]** (“The invention may be embodied in various forms. One embodiment includes a mobile device, such as a personal digital assistant (PDA), mobile terminal, cellular telephone or the like, having a navigation display of an organizational model. The organizational model graphically represents features of a computer program for the device. Referring to FIG. 1B, one embodiment of a mobile device 110 on which the present invention may be utilized generally includes a display 112 and a keypad 118. The keypad 118 may include a left movement button 120, a right movement button 122, an up arrow button 124, a down arrow button 126, an input wheel 128, and other input keys 130. The keys 120, 122, 124 and 126 could also comprise soft keys, the function of which adapts to the status of the user interface. The input wheel 128 can rotate for providing rotary inputs to the device and can be depressed as a whole, like a button, for providing selector

inputs. In another embodiment, the input wheel is located on the keypad as a rotatable button that can be rotated clockwise and counterclockwise, as well as depressed as a whole.”);

- **Aaltonen, [0060]** (“In one embodiment, referring to FIG. 2A., each summary area 210, also referred to herein as a “stripe” provides an overview to the events and objects of a selected category 210A-210F. Each summary area category 210A-210F could also include one or more indicators or icons 230. The icons could include for example text, images, dynamic icons, hypertext and user interface widgets. The indicators or icons 230 generally provide visual links to other information related to the summary area category. The types of indicators 230 can be any suitable type, number or combination thereof, depending only on the particular application and device.”);
- **Aaltonen, [0121]** (“FIG. 22G illustrates an example of a notifier 2280 pertaining to the content category 2206. In this example, the indication or icon 2281 could indicate the presence of a pending or available action, such as for example an “MP3” playback. In the example shown in FIG. 22G, the icon associated with the category 2206 is highlighted to indicate the presence or activation of an event. The icon 2281 is displayed to represent the subcategory or the nature or type of event. In one embodiment, the color could change for example, to correspond to the media type, the user being aware of or able to determine the media type from the icon or indication. The content notification icon 2280 in FIG. 22G is shown in a minimal or minimized state. The icon 2280 could remain displayed as long as the content remains available or some other action by the user takes place.”);
- **Aaltonen, [0158]** (“In one embodiment, the present invention provides a user interface for accessing, consuming, managing and sharing digital content between multiple connected devices. Many types of devices can be used for getting, creating, consuming, sharing and managing digital content. Some of these devices can include for example, DVB-T/H receivers, game consoles, PC’s, cameras, MP3 players, smart phones, PDA’s and mobile telephones. When these devices are connected or interconnected in some way, they can form a device ecosystem, such as that shown in FIG. 25A. The embodiments of the present invention provide a user interface to access, consume, manage and share digital content between multiple connected devices.”).

137. In addition to Paasonen's teachings, as discussed in Section IX.A.2(k), Aaltonen discloses the portable terminal executes an "MP3" playback" function. Aaltonen [0121]. As discussed in Section IX.C.1, a person of ordinary skill in the art would have been motivated to modify Paasonen's music player with Aaltonen's MP3 teachings in order to implement the player with a common music file format.

(b) Claim [7]

138. **Claim [7] recites "The method of claim 3, wherein the third command and the fifth command are inputted through at least one of a predetermined key, a user's voice, and a motion of the mobile terminal."** This element is met by Paasonen (or Paasonen in view of Nakano) in further view of Aaltonen.

139. In addition to the teachings from Paasonen that I discussed above in Section IX., **Aaltonen discloses commands are inputted through a user's voice** (*e.g.*, "voice command ability"). *See, e.g.*, Aaltonen (emphasis added):

- **Aaltonen, [0053]** ("The user interface of the present invention is generally provided on a display of an electronic device, such as for example, a portable terminal device, including a mobile telephone. FIG. 1A illustrates a schematic diagram of one embodiment of an electronic device 100 incorporating features of the present invention. The device 100 can be a wireless terminal device operating in a communication system such as for example, a GSM, GPRS, UMTS or Bluetooth system. The electronic device can also be for example a hand-held, portable or desktop computer or a game device or console, or a personal digital assistant ("PDA"). In alternate

embodiments, the user interface of the present invention can be implemented on any content and task driven electronic device. The device 100 generally comprises a processor 101 and a memory 102 for operating tasks of the device 100 and for running applications 103 stored in the device. For operating in a communications system, the device can include a transceiver 104 and an antenna 105. For data input, the device 100 can include a keyboard 106, which can be a soft key keyboard or touch sensitive area on a surface of the device providing visual symbols or other indicators for the user to select by touching the symbol. The device 100 can also include other input means, such as for example a joystick, a rocker switch controller, a touch sensitive display or voice command ability. For data output, the device 100 includes a display 107 that can be a monochrome or color display, an LCD panel, a touch sensitive panel or other suitable display as well as vibration motors for tactile output. In one embodiment, the processor 101 can accommodate other input/output devices 109 as well as a speaker/microphone 108.”).

140. In addition to Paasonen teachings, as I discussed above in Section IX.A.2(n), Aaltonen discloses accepting user input through the terminal’s “voice command ability.” Aaltonen [0053]. As discussed in Section IX.C.1, a person of ordinary skill in the art would have been motivated to modify Paasonen’s user input mechanism with Aaltonen’s voice command teachings in order to permit a user to operate Paasonen’s mobile terminal hands-free. By implementing the device hands-free with voice, the user can then control how to interact with menus on the device, including accessing menus and closing applications as necessary.

D. Ground 5: Hotelling Anticipates and Renders Obvious Claims 1-4, and 7

1. Overview of Hotelling

141. Hotelling is titled “Multi-functional hand-held device” and it incorporates both provisional App. No. 60/658,777 (“Hotelling-Prov.,” Ex. 1010) and U.S. Pub. No. 2006/0026521 (“Hotelling-521”) by reference. Hotelling [0001]. Hotelling-Prov. also independently incorporates Hotelling-521 by reference. *See* Hotelling-Prov. 1:8-22. Hotelling teaches a “hand-held electronic device” that is capable of “simultaneously” running multiple functions such as a “Music Player” and “Cell Phone.” Hotelling-Prov. 7:19-34, 20:9-21. Figure 18 shows an embodiment of the menu presented to the user in order to select such functions:

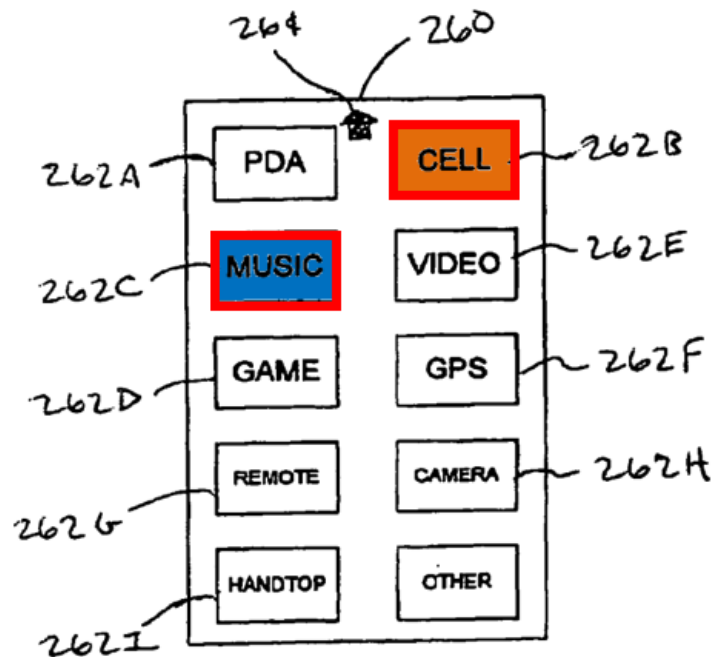


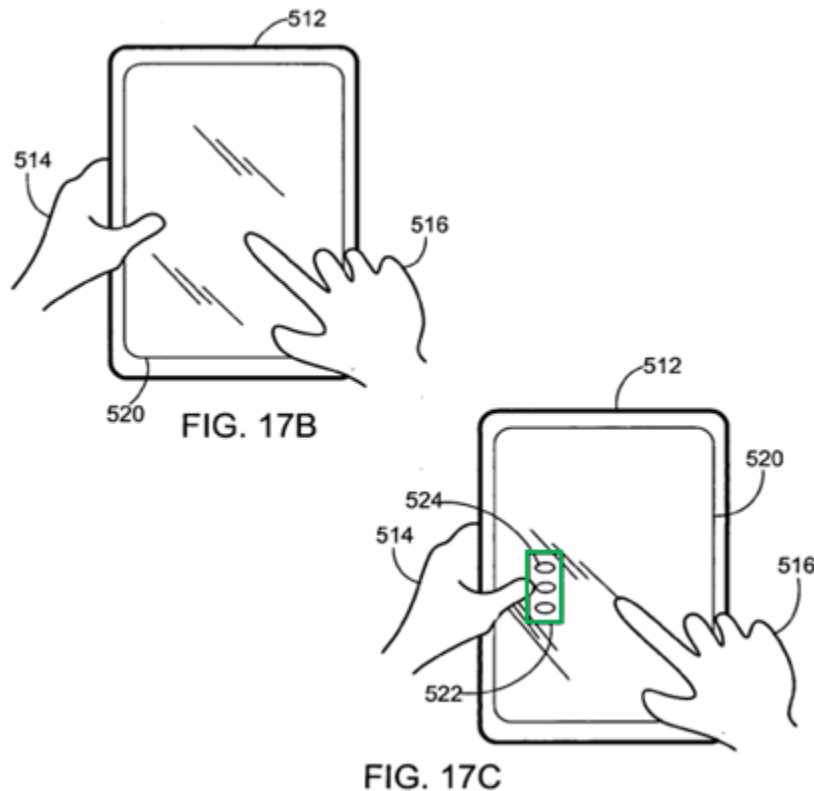
FIG. 18

Hotelling-Prov., Fig. 18.

142. By selecting any of the icons, which are presented in **designated locations** on a “main menu GUI” the user can launch any of several functions available on the device, such as a “**music player**,” “**cell phone**,” “game player,” “video player,” and so on. Hotelling-Prov., 18:19-29. When launched, the “GUI” corresponding to a particular function is “activate[d]” and displayed when the user selects the icon for that function from the “main menu GUI.” Hotelling-Prov. 18:10-29. The user can return to the main menu at any point when a new function is desired to “flip (or scroll) through the various GUIs until the desired GUI is found.” Hotelling-Prov., 18:19-29, 18:30-33. Furthermore, a user can display the GUI for a second function in a “split screen” mode by issuing a command from the

main menu. Hotelling-Prov. 20:9-21. For example, a user may activate this mode by “simultaneously touch[ing] two device icons.” Hotelling-Prov. 20:9-21.

143. Hotelling teaches that, in addition to selecting controls that are displayed as part of the GUIs to the user, contextual “floating controls” can be activated by touching the screen in certain areas and with certain gestures. Hotelling-521, Figs. 6A, 17A-17E, [0070], [0086]-[0087], [0111]-[0114]. In Hotelling-521 Figures 17B–C, for example, “a portion of the [user’s] thumb . . . positioned over the touchscreen” in Figure 17B causes a “control box” to be displayed “adjacent the thumb”:



Hotelling-521, Figs. 17B–17C, [0114]. The contextual controls are presented as floating controls to the user and can include “menus,” a “volume knob,” and/or “play, pause” “buttons” for controlling—e.g., “music” applications. Hotelling-521 [0070], [0112]-[0113].

2. Hotelling’s Effective Priority Date and Incorporation of Hotelling-521

144. I understand that Hotelling incorporates by reference and claim priority to App. 60/658777 (“Hotelling-Prov.”), filed on March 4, 2005, which is earlier than the earliest possible priority date of the ’631 Patent. Additionally, both Hotelling and Hotelling-Prov. each incorporate by reference Application No. 10/903,964, published as US2006/0026521 (“Hotelling-521”), making Hotelling-521 part of both the Hotelling and Hotelling-Prov disclosures. See Hotelling [0002]; Hotelling-Prov. 1:8-22.

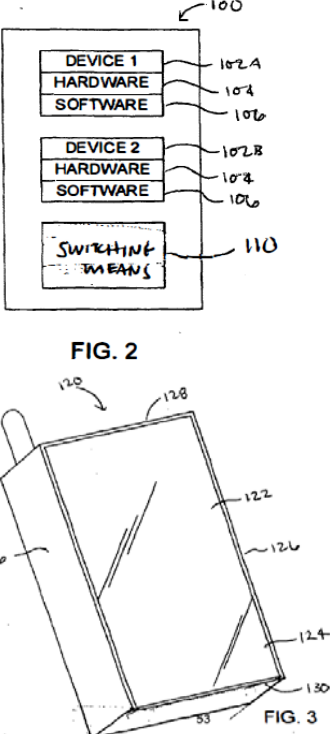
145. In forming my opinions, I was instructed to treat Hotelling-521 as part of both Hotelling’s disclosure and of Hotelling-Prov.’s disclosure. But even if Hotelling-521 were considered a separate reference, a person of ordinary skill in the art would have been motivated to modify Hotelling’s and Hotelling-Prov.’s teachings with Hotelling-521’s teachings in light of the explicit incorporation of Hotelling-521 into Hotelling and Hotelling-Prov. Moreover, a person of ordinary skill in the art would have found it straightforward to apply Hotelling-521’s

teachings to Hotelling and Hotelling-Prov., and would have had a reasonable expectation of success in doing so. My analysis therefore would not change if Hotelling-521 were treated as a separate reference distinct from Hotelling and Hotelling-Prov.

146. In forming my opinions, I was asked to consider whether Hotelling is entitled to claim priority to the filing date of Hotelling-Prov. That includes considering whether Hotelling-Prov. provides written description and enablement support for at least one claim in Hotelling, as well as written description and enablement support for the portions of Hotelling that I rely upon for my opinions. It is my opinion that these criteria are satisfied, and Hotelling is entitled to the priority date of Hotelling-Prov.

147. First, having reviewed the intrinsic evidence, it is my opinion that Hotelling-Prov. provides enabling written description support for at least 1 claim in Hotelling. In the table below, I identify example disclosures from Hotelling-Prov. that provide enabling written description support for Hotelling. I also note that the Hotelling-Prov. specification is largely identical to Hotelling specification, and for every paragraph in the Hotelling specification there is an equivalent disclosure in the Hotelling-Prov. In preparing this table, it was unnecessary for me to construe any of the claim terms in Hotelling or to determine whether or not the preamble from claim 1 of Hotelling was limiting, because under any reasonable

interpretation of the claim, Hotelling-Prov. provides more than sufficient enabling written description support. I reserve the right to consider the proper construction of claim 1 of Hotelling or supplement this table if asked to do so at a later time.

Hotelling	Hotelling-Prov.—e.g.:
[1.pre] A hand-held electronic device, comprising:	Figs. 2-3:  4:8-12 (“Fig. 2 is a simplified diagram of a multi-functional hand-held device, in accordance with one embodiment of the present invention. Fig. 3 is a perspective view of a substantially full screen hand-held device with limited buttons, in accordance with one embodiment of the present invention.”)
[1.a] a multi-touch input surface; and	23:6-9 (“In accordance with another embodiment, a hand-held electronic device may include a touch sensitive surface that provides either a larger surface for tracking touch inputs or smaller dedicated areas

Hotelling	Hotelling-Prov.—e.g.:
	<i>such as touch buttons</i> for performing dedicated functions.”) 24:1-3 (“In one particular implementation, <i>the touch sensitive surface is a multipoint capacitive touch device</i> that is divided into several independent and spatially distinct sensing points, nodes or regions that are positioned throughout the device.”).
[1.b] a <i>processing unit operatively connected to said multi-touch input surface,</i>	Fig. 28.: 23:6-9 (“In accordance with another embodiment, a <i>hand-held electronic device may include a touch sensitive surface</i> that provides either a larger surface <i>for tracking touch inputs</i> or smaller dedicated areas such as touch buttons for performing dedicated functions.”) 24:1-3 (“In one particular implementation, <i>the touch sensitive surface is a multipoint capacitive touch device</i> that is divided into several independent and spatially distinct sensing points, nodes or regions that

Hotelling	Hotelling-Prov.— <i>e.g.</i> :
	are positioned throughout the device.”). 47:31-48:5 (“10. OVERALL BLOCK DIAGRAM Fig. 28 is a block diagram of an exemplary hand-held device 600 in accordance with one embodiment of the present invention. <i>The hand-held device 600 typically includes a controller 602 (e.g., CPU) configured to execute instructions and to carry out operations associated with the hand-held device.</i> For example, using instructions retrieved for example from memory, the controller 602 may control the reception and manipulation of input and output data between components of the hand-held device 600.”) 48:28-49:3 (“<i>The hand-held device 600 also includes various input devices 606 that are operatively coupled to the controller 602.</i> The input devices 606 are configured to transfer data from the outside world into the hand-held device 600. As shown, the input devices 606 may correspond to both data entry mechanisms and data capture mechanisms. <i>In particular, the input devices 606 may include touch sensing devices 608 such as touch screens, touch pads and touch sensing surfaces,</i> mechanical actuators 610 such as button or wheels or hold switches (611), motion sensing devices”)
[1.d] said processing unit capable of <i>receiving a plurality of concurrent touch inputs from a user via said multi-touch input surface and discriminating a user requested action from the touch inputs;</i> and	20:9-21 (“C. OPERATING AT LEAST TWO FUNCTIONALITIES SIMULTANEOUSLY In one embodiment, the device instead of activating one device, may also activate two devices (or more). In cases such as these, the software for both devices is activated and the display operates in a split screen mode where the screen is parsed into different sections, each section including the device GUI. By way of example, in a landscape mode, a first device GUI may be presented on the left side of the display screen and a second device GUI may be presented on the right side of the display screen. In cases such as this, the GUI

Hotelling	Hotelling-Prov.—e.g.:																																																																																	
	would be reduced to accommodate the new space. The dual mode may be selected in a variety of ways. <i>In one implementation, when the user simultaneously touches two device icons, the device activates both device functionalities and brings both GUIs into view on the screen.</i>”) Figs. 27A-27E: <table><tr><th>#</th><th>UI Mode</th><th>Light Touch or Tactile Click</th><th># Fingers used</th><th>Movement during touch or click</th><th>Duration</th><th>Touchdown location (virtual button?)</th><th>Proposed Behavior</th><th>Feedback</th></tr><tr><td colspan="9">Basic list navigation</td></tr><tr><td>List Nav.</td><td>light</td><td></td><td>1</td><td>none</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On lift-off, highlight item(if any) under touch</td><td>highlight item touched</td></tr><tr><td>List Nav.</td><td>click</td><td></td><td>1</td><td>NONE</td><td>short</td><td>virtual button / list item</td><td>On touch-down, do nothing. On click-down highlight item under touch (if any). On un-click, execute the item clicked</td><td>animate screen sliding RIGHT to expose new screen</td></tr><tr><td>List Nav.</td><td>click</td><td></td><td>1</td><td>NONE</td><td>short</td><td>PLAY / PAUSE virtual button</td><td>On touch-down, do nothing. On click-down highlight Play/Pause icon. On un-click, PLAY/PAUSE</td><td>Change PLAY/PAUSE icon</td></tr><tr><td>List Nav.</td><td>click</td><td></td><td>1</td><td>NONE</td><td>LONG</td><td>virtual button</td><td>On touch-down, do nothing. On click-down highlight item under touch (if any). After delay, bring-up context sensitive menu.</td><td>Context sensitive menu pop-up</td></tr><tr><td>List Nav.</td><td>light or click</td><td></td><td>1</td><td>UP/DOWN</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. Scroll list UP/DOWN depending on UP/DOWN distance moved, use inertia & brakes, disregard LEFT/RIGHT motion during gesture. If highlighted item slides off visible screen, change highlighted item to one that is visible.</td><td>list scrolling UP/DOWN, first few letters enlarged</td></tr><tr><td>List Nav.</td><td>light or click</td><td></td><td>1</td><td>LEFT</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On left motion, open highlighted folder or execute highlighted item (like legacy SELECT button)</td><td>animate screen sliding LEFT</td></tr><tr><td>List Nav.</td><td>light or click</td><td></td><td>1</td><td>RIGHT</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)</td><td>animate screen sliding RIGHT</td></tr></table> FIG. 27A	#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback	Basic list navigation									List Nav.	light		1	none	any	anywhere	On touch-down, do nothing. On lift-off, highlight item(if any) under touch	highlight item touched	List Nav.	click		1	NONE	short	virtual button / list item	On touch-down, do nothing. On click-down highlight item under touch (if any). On un-click, execute the item clicked	animate screen sliding RIGHT to expose new screen	List Nav.	click		1	NONE	short	PLAY / PAUSE virtual button	On touch-down, do nothing. On click-down highlight Play/Pause icon. On un-click, PLAY/PAUSE	Change PLAY/PAUSE icon	List Nav.	click		1	NONE	LONG	virtual button	On touch-down, do nothing. On click-down highlight item under touch (if any). After delay, bring-up context sensitive menu.	Context sensitive menu pop-up	List Nav.	light or click		1	UP/DOWN	any	anywhere	On touch-down, do nothing. Scroll list UP/DOWN depending on UP/DOWN distance moved, use inertia & brakes, disregard LEFT/RIGHT motion during gesture. If highlighted item slides off visible screen, change highlighted item to one that is visible.	list scrolling UP/DOWN, first few letters enlarged	List Nav.	light or click		1	LEFT	any	anywhere	On touch-down, do nothing. On left motion, open highlighted folder or execute highlighted item (like legacy SELECT button)	animate screen sliding LEFT	List Nav.	light or click		1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT
#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback																																																																										
Basic list navigation																																																																																		
List Nav.	light		1	none	any	anywhere	On touch-down, do nothing. On lift-off, highlight item(if any) under touch	highlight item touched																																																																										
List Nav.	click		1	NONE	short	virtual button / list item	On touch-down, do nothing. On click-down highlight item under touch (if any). On un-click, execute the item clicked	animate screen sliding RIGHT to expose new screen																																																																										
List Nav.	click		1	NONE	short	PLAY / PAUSE virtual button	On touch-down, do nothing. On click-down highlight Play/Pause icon. On un-click, PLAY/PAUSE	Change PLAY/PAUSE icon																																																																										
List Nav.	click		1	NONE	LONG	virtual button	On touch-down, do nothing. On click-down highlight item under touch (if any). After delay, bring-up context sensitive menu.	Context sensitive menu pop-up																																																																										
List Nav.	light or click		1	UP/DOWN	any	anywhere	On touch-down, do nothing. Scroll list UP/DOWN depending on UP/DOWN distance moved, use inertia & brakes, disregard LEFT/RIGHT motion during gesture. If highlighted item slides off visible screen, change highlighted item to one that is visible.	list scrolling UP/DOWN, first few letters enlarged																																																																										
List Nav.	light or click		1	LEFT	any	anywhere	On touch-down, do nothing. On left motion, open highlighted folder or execute highlighted item (like legacy SELECT button)	animate screen sliding LEFT																																																																										
List Nav.	light or click		1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT																																																																										

Hotelling	Hotelling-Prov.—e.g.:																																																																																																																																																																											
	<table><tr><th>#</th><th>UI Mode</th><th>Light Touch or Tactile Click</th><th># Fingers used</th><th>Movement during touch or click</th><th>Duration</th><th>Touchdown location (virtual button?)</th><th>Proposed Behavior</th><th>Feedback</th></tr><tr><td colspan="9">2D photo thumbnail preview / navigation</td></tr><tr><td></td><td>photo preview</td><td>light</td><td>1</td><td>none</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On lift-off, highlight item (if any) under touch</td><td>highlight item touched</td></tr><tr><td></td><td>photo preview</td><td>click</td><td>1</td><td>NONE</td><td>short</td><td>virtual button / thumbnail</td><td>On touch-down, do nothing. On click-down highlight item under touch (if any). On un-click, execute the item clicked</td><td>animate screen sliding RIGHT to expose new screen</td></tr><tr><td></td><td>photo preview</td><td>click</td><td>1</td><td>NONE</td><td>short</td><td>PLAY / PAUSE virtual button</td><td>On touch-down, do nothing. On click-down highlight Play/Pause icon. On un-click, PLAY/PAUSE</td><td>Change PLAY/PAUSE icon</td></tr><tr><td></td><td>photo preview</td><td>click</td><td>1</td><td>NONE</td><td>LONG</td><td>virtual button</td><td>On touch-down, do nothing. On click-down highlight item under touch (if any). After delay, bring-up context sensitive menu.</td><td>Context sensitive menu pop-up</td></tr><tr><td></td><td>photo preview</td><td>light or click</td><td>1</td><td>UP/DOWN</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. Scroll list of thumbnails UP/DOWN depending on UP/DOWN distance moved, use inertia & brakes, disregard LEFT/RIGHT motion during gesture. If highlighted item slides off visible screen, change highlighted item to one that is visible.</td><td>list of thumbnails scrolling UP/DOWN</td></tr><tr><td></td><td>photo preview</td><td>light or click</td><td>1</td><td>LEFT</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On left motion, open highlighted folder or execute highlighted item (like legacy SELECT button)</td><td>animate screen sliding LEFT</td></tr><tr><td></td><td>photo preview</td><td>light or click</td><td>1</td><td>RIGHT</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)</td><td>animate screen sliding RIGHT</td></tr></table> FIG. 27B <table><tr><th>#</th><th>UI Mode</th><th>Light Touch or Tactile Click</th><th># Fingers used</th><th>Movement during touch or click</th><th>Duration</th><th>Touchdown location (virtual button?)</th><th>Proposed Behavior</th><th>Feedback</th></tr><tr><td colspan="9">Now Playing</td></tr><tr><td></td><td>Now Playing</td><td>light</td><td>1</td><td>none</td><td>any</td><td>anywhere</td><td>do nothing</td><td>n/a</td></tr><tr><td></td><td>Now Playing</td><td>click</td><td>1</td><td>NONE</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On click-down PLAY / PAUSE</td><td>change PLAY/PAUSE icon</td></tr><tr><td></td><td>Now Playing</td><td>light</td><td>1</td><td>UP/DOWN</td><td>any</td><td>anywhere</td><td>do nothing</td><td>n/a</td></tr><tr><td></td><td>Now Playing</td><td>light</td><td>1</td><td>LEFT</td><td>any</td><td>anywhere</td><td>do nothing</td><td>n/a</td></tr><tr><td></td><td>Now Playing</td><td>light</td><td>1</td><td>RIGHT</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)</td><td>animate screen sliding RIGHT</td></tr><tr><td></td><td>Now Playing</td><td>click</td><td>1</td><td>UP/DOWN</td><td>any</td><td>anywhere</td><td>On touch-down & click-down, do nothing. On UP/DOWN motion detect, increase/decrease volume and disregard LEFT/RIGHT motion</td><td>Volume slider moves UP/DOWN</td></tr><tr><td></td><td>Now Playing</td><td>click</td><td>1</td><td>LEFT</td><td>any</td><td>anywhere</td><td>On touch-down & click-down, do nothing. On left motion, go to next track (like legacy NEXT button)</td><td>animate screen sliding LEFT</td></tr><tr><td></td><td>Now Playing</td><td>click</td><td>1</td><td>RIGHT</td><td>any</td><td>anywhere</td><td>On touch-down & click-down, do nothing. On right motion detect, go to prev. track (like legacy PREV button)</td><td>animate screen sliding RIGHT</td></tr></table> FIG. 27C	#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback	2D photo thumbnail preview / navigation										photo preview	light	1	none	any	anywhere	On touch-down, do nothing. On lift-off, highlight item (if any) under touch	highlight item touched		photo preview	click	1	NONE	short	virtual button / thumbnail	On touch-down, do nothing. On click-down highlight item under touch (if any). On un-click, execute the item clicked	animate screen sliding RIGHT to expose new screen		photo preview	click	1	NONE	short	PLAY / PAUSE virtual button	On touch-down, do nothing. On click-down highlight Play/Pause icon. On un-click, PLAY/PAUSE	Change PLAY/PAUSE icon		photo preview	click	1	NONE	LONG	virtual button	On touch-down, do nothing. On click-down highlight item under touch (if any). After delay, bring-up context sensitive menu.	Context sensitive menu pop-up		photo preview	light or click	1	UP/DOWN	any	anywhere	On touch-down, do nothing. Scroll list of thumbnails UP/DOWN depending on UP/DOWN distance moved, use inertia & brakes, disregard LEFT/RIGHT motion during gesture. If highlighted item slides off visible screen, change highlighted item to one that is visible.	list of thumbnails scrolling UP/DOWN		photo preview	light or click	1	LEFT	any	anywhere	On touch-down, do nothing. On left motion, open highlighted folder or execute highlighted item (like legacy SELECT button)	animate screen sliding LEFT		photo preview	light or click	1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT	#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback	Now Playing										Now Playing	light	1	none	any	anywhere	do nothing	n/a		Now Playing	click	1	NONE	any	anywhere	On touch-down, do nothing. On click-down PLAY / PAUSE	change PLAY/PAUSE icon		Now Playing	light	1	UP/DOWN	any	anywhere	do nothing	n/a		Now Playing	light	1	LEFT	any	anywhere	do nothing	n/a		Now Playing	light	1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT		Now Playing	click	1	UP/DOWN	any	anywhere	On touch-down & click-down, do nothing. On UP/DOWN motion detect, increase/decrease volume and disregard LEFT/RIGHT motion	Volume slider moves UP/DOWN		Now Playing	click	1	LEFT	any	anywhere	On touch-down & click-down, do nothing. On left motion, go to next track (like legacy NEXT button)	animate screen sliding LEFT		Now Playing	click	1	RIGHT	any	anywhere	On touch-down & click-down, do nothing. On right motion detect, go to prev. track (like legacy PREV button)	animate screen sliding RIGHT
#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback																																																																																																																																																																				
2D photo thumbnail preview / navigation																																																																																																																																																																												
	photo preview	light	1	none	any	anywhere	On touch-down, do nothing. On lift-off, highlight item (if any) under touch	highlight item touched																																																																																																																																																																				
	photo preview	click	1	NONE	short	virtual button / thumbnail	On touch-down, do nothing. On click-down highlight item under touch (if any). On un-click, execute the item clicked	animate screen sliding RIGHT to expose new screen																																																																																																																																																																				
	photo preview	click	1	NONE	short	PLAY / PAUSE virtual button	On touch-down, do nothing. On click-down highlight Play/Pause icon. On un-click, PLAY/PAUSE	Change PLAY/PAUSE icon																																																																																																																																																																				
	photo preview	click	1	NONE	LONG	virtual button	On touch-down, do nothing. On click-down highlight item under touch (if any). After delay, bring-up context sensitive menu.	Context sensitive menu pop-up																																																																																																																																																																				
	photo preview	light or click	1	UP/DOWN	any	anywhere	On touch-down, do nothing. Scroll list of thumbnails UP/DOWN depending on UP/DOWN distance moved, use inertia & brakes, disregard LEFT/RIGHT motion during gesture. If highlighted item slides off visible screen, change highlighted item to one that is visible.	list of thumbnails scrolling UP/DOWN																																																																																																																																																																				
	photo preview	light or click	1	LEFT	any	anywhere	On touch-down, do nothing. On left motion, open highlighted folder or execute highlighted item (like legacy SELECT button)	animate screen sliding LEFT																																																																																																																																																																				
	photo preview	light or click	1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT																																																																																																																																																																				
#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback																																																																																																																																																																				
Now Playing																																																																																																																																																																												
	Now Playing	light	1	none	any	anywhere	do nothing	n/a																																																																																																																																																																				
	Now Playing	click	1	NONE	any	anywhere	On touch-down, do nothing. On click-down PLAY / PAUSE	change PLAY/PAUSE icon																																																																																																																																																																				
	Now Playing	light	1	UP/DOWN	any	anywhere	do nothing	n/a																																																																																																																																																																				
	Now Playing	light	1	LEFT	any	anywhere	do nothing	n/a																																																																																																																																																																				
	Now Playing	light	1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT																																																																																																																																																																				
	Now Playing	click	1	UP/DOWN	any	anywhere	On touch-down & click-down, do nothing. On UP/DOWN motion detect, increase/decrease volume and disregard LEFT/RIGHT motion	Volume slider moves UP/DOWN																																																																																																																																																																				
	Now Playing	click	1	LEFT	any	anywhere	On touch-down & click-down, do nothing. On left motion, go to next track (like legacy NEXT button)	animate screen sliding LEFT																																																																																																																																																																				
	Now Playing	click	1	RIGHT	any	anywhere	On touch-down & click-down, do nothing. On right motion detect, go to prev. track (like legacy PREV button)	animate screen sliding RIGHT																																																																																																																																																																				

Hotelling	Hotelling-Prov.—e.g.:																																																																																																																					
	<table><tr><th>#</th><th>UI Mode</th><th>Light Touch or Tactile Click</th><th># Fingers used</th><th>Movement during touch or click</th><th>Duration</th><th>Touchdown location (virtual button?)</th><th>Proposed Behavior</th><th>Feedback</th></tr><tr><td colspan="9">Full-screen Item (Photo, map view, calendar day, single contact view, etc.)</td></tr><tr><td>Full-screen item</td><td></td><td>light</td><td>1</td><td>none</td><td>any</td><td>anywhere</td><td>do nothing</td><td>n/a</td></tr><tr><td>Full-screen item</td><td></td><td>click</td><td>1</td><td>NONE</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On click-down PLAY / PAUSE.</td><td>change PLAY/PAUSE icon</td></tr><tr><td>Full-screen item</td><td></td><td>light</td><td>1</td><td>UP/DOWN</td><td>any</td><td>anywhere</td><td>do nothing</td><td>n/a</td></tr><tr><td>Full-screen item</td><td></td><td>light</td><td>1</td><td>LEFT</td><td>any</td><td>anywhere</td><td>do nothing</td><td>n/a</td></tr><tr><td>Full-screen item</td><td></td><td>light</td><td>1</td><td>RIGHT</td><td>any</td><td>anywhere</td><td>On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)</td><td>animate screen sliding RIGHT</td></tr><tr><td>Full-screen item</td><td></td><td>click</td><td>1</td><td>UP/DOWN</td><td>any</td><td>anywhere</td><td>On touch-down & click-down, do nothing. On UP/DOWN motion detect, increase/decrease volume and disregard LEFT/RIGHT motion</td><td>Volume slider moves UP/DOWN</td></tr><tr><td>Full-screen item</td><td></td><td>click</td><td>1</td><td>LEFT</td><td>any</td><td>anywhere</td><td>On touch-down & click-down, do nothing. On left motion, go to next item (like legacy NEXT button)</td><td>animate screen sliding LEFT</td></tr><tr><td>Full-screen item</td><td></td><td>click</td><td>1</td><td>RIGHT</td><td>any</td><td>anywhere</td><td>On touch-down & click-down, do nothing. On right motion detect, go to prev. item (like legacy PREV button)</td><td>animate screen sliding RIGHT</td></tr></table> FIG. 27D <table><tr><th>#</th><th>UI Mode</th><th>Light Touch or Tactile Click</th><th># Fingers used</th><th>Movement during touch or click</th><th>Duration</th><th>Touchdown location (virtual button?)</th><th>Proposed Behavior</th><th>Feedback</th></tr><tr><td colspan="9">Optional Multi-touch gestures on full-screen item</td></tr><tr><td>Full-screen item</td><td></td><td>light or click</td><td>2 or more</td><td>Up, down, right, left</td><td>any</td><td>anywhere</td><td>translate image according total incremental centroid translation of contacts, scale (zoom) image according to total incremental spread (radius of gyration) of contacts, rotate image according to incremental rotation of contacts</td><td>image translation, scaling and rotation</td></tr></table> FIG. 27E	#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback	Full-screen Item (Photo, map view, calendar day, single contact view, etc.)									Full-screen item		light	1	none	any	anywhere	do nothing	n/a	Full-screen item		click	1	NONE	any	anywhere	On touch-down, do nothing. On click-down PLAY / PAUSE.	change PLAY/PAUSE icon	Full-screen item		light	1	UP/DOWN	any	anywhere	do nothing	n/a	Full-screen item		light	1	LEFT	any	anywhere	do nothing	n/a	Full-screen item		light	1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT	Full-screen item		click	1	UP/DOWN	any	anywhere	On touch-down & click-down, do nothing. On UP/DOWN motion detect, increase/decrease volume and disregard LEFT/RIGHT motion	Volume slider moves UP/DOWN	Full-screen item		click	1	LEFT	any	anywhere	On touch-down & click-down, do nothing. On left motion, go to next item (like legacy NEXT button)	animate screen sliding LEFT	Full-screen item		click	1	RIGHT	any	anywhere	On touch-down & click-down, do nothing. On right motion detect, go to prev. item (like legacy PREV button)	animate screen sliding RIGHT	#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback	Optional Multi-touch gestures on full-screen item									Full-screen item		light or click	2 or more	Up, down, right, left	any	anywhere	translate image according total incremental centroid translation of contacts, scale (zoom) image according to total incremental spread (radius of gyration) of contacts, rotate image according to incremental rotation of contacts	image translation, scaling and rotation
#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback																																																																																																														
Full-screen Item (Photo, map view, calendar day, single contact view, etc.)																																																																																																																						
Full-screen item		light	1	none	any	anywhere	do nothing	n/a																																																																																																														
Full-screen item		click	1	NONE	any	anywhere	On touch-down, do nothing. On click-down PLAY / PAUSE.	change PLAY/PAUSE icon																																																																																																														
Full-screen item		light	1	UP/DOWN	any	anywhere	do nothing	n/a																																																																																																														
Full-screen item		light	1	LEFT	any	anywhere	do nothing	n/a																																																																																																														
Full-screen item		light	1	RIGHT	any	anywhere	On touch-down, do nothing. On right motion, go back out a folder (like legacy MENU function)	animate screen sliding RIGHT																																																																																																														
Full-screen item		click	1	UP/DOWN	any	anywhere	On touch-down & click-down, do nothing. On UP/DOWN motion detect, increase/decrease volume and disregard LEFT/RIGHT motion	Volume slider moves UP/DOWN																																																																																																														
Full-screen item		click	1	LEFT	any	anywhere	On touch-down & click-down, do nothing. On left motion, go to next item (like legacy NEXT button)	animate screen sliding LEFT																																																																																																														
Full-screen item		click	1	RIGHT	any	anywhere	On touch-down & click-down, do nothing. On right motion detect, go to prev. item (like legacy PREV button)	animate screen sliding RIGHT																																																																																																														
#	UI Mode	Light Touch or Tactile Click	# Fingers used	Movement during touch or click	Duration	Touchdown location (virtual button?)	Proposed Behavior	Feedback																																																																																																														
Optional Multi-touch gestures on full-screen item																																																																																																																						
Full-screen item		light or click	2 or more	Up, down, right, left	any	anywhere	translate image according total incremental centroid translation of contacts, scale (zoom) image according to total incremental spread (radius of gyration) of contacts, rotate image according to incremental rotation of contacts	image translation, scaling and rotation																																																																																																														
[1.e] <i>a display device operatively coupled to the processing unit and configured to present a user</i>	18:10-17 (“Before a particular device of the multi-functional device can be used by a user, it is typically selected for use. The selection can come in a variety of forms. In one implementation, the selection is made via a main menu that includes soft buttons or icons that,																																																																																																																					

Hotelling	Hotelling-Prov.— <i>e.g.</i> :
<i>interface.</i>	when selected, activate the device functionality associated with the soft button. <i>During activation, the GUI for that particular device is brought into view on the display (see Figs. 9-17) and the software associated with the device is installed, loaded or woken up.</i> From that point on, the multi-functional device operates like the selected device.”). 49:5-9 (“<i>The hand-held device 600 also includes various output devices 620 that are operatively coupled to the controller 602.</i> The output devices 620 are configured to transfer data from the hand-held device 600 to the outside world. <i>The output devices 620 may include a display 622 such as an LCD,</i> speakers or jacks 624, audio/tactile feedback devices 626, light indicators 628, and the like”)

148. Second, it is my opinion that Hotelling-Prov. provides enabling written description support for the portions of Hotelling that I substantively rely upon in formulating my opinion that Hotelling anticipates or renders obvious—alone, in view of Hellyar, or in view of Hellyar and Tu—the Challenged claims. I have prepared a redline document that compares the passages of Hotelling with Hotelling-Prov. Ex. 1033. Based on my review of this comparison document, Hotelling and Hotelling Prov are substantively identical, and almost every paragraph in Hotelling has an equivalent corresponding disclosure in Hotelling-Prov. I have also prepared the table below to illustrate a mapping passages of Hotelling-Prov. relied on in the grounds below with equivalent paragraphs from

Hotelling. Accordingly, as set forth in this section, Hotelling-Prov. provides enabling written description support for each claim in Hotelling, as well as the particular disclosures from Hotelling that anticipate or render obvious the Claims.

Hotelling-Prov. Disclosure	Corresponding Hotelling Disclosure
Figs. 1-28	Figs. 1-28
7:25-33	[0041]
8:29-9:2	[0046]
9:12-23	[0048]
15:1-15	[0070]
16:8-16	[0073]
16:28-31	[0076]
16:32-17:5	[0077]
18:19-29	[0085]
18:31-19:7	[0086]
19:9-16	[0087]
19:18-25	[0088]
19:27-30	[0089]
20:9-21	[0092]
20:25-21:2	[0093]

Hotelling-Prov. Disclosure	Corresponding Hotelling Disclosure
22:6-20	[0097]
33:5-12	[0127]
34:15-23	[0133]
47:31-48:10	[0170]
48:5-9	[0173]
48:12-26	[0171]
48:28-49:3	[0172]
49:11-15	[0174]

149. Based on the disclosures in the provisional cited herein and in the table above comparing claim 1 of Hotelling with Hotelling-Prov., a person of ordinary skill in the art at the time (relative to the Hotelling reference) would have been able to practice the claim without undue experimentation. By that timeframe, the concepts encompassed by Hotelling were well-understood in the field.

3. Invalidity of Claims 1–4 and 7 over Hotelling

150. As I discussed when introducing the grounds of unpatentability in Sections IX, and the overview of Hotelling in IX.D.1, Hotelling incorporates by reference and is substantially the same as Hotelling-Prov, and both Hotelling and Hotelling-Prov each incorporate by reference Hotelling-521. See Hotelling [0001]-

[0002]; Hotelling-Prov. 1:16-18. My citations below to Hotelling-Prov. should be understood to also cite and rely upon the corresponding Hotelling disclosures identified in the table above. See Section IX.D.2.

(a) Element [1.pre]

151. **Element [1.pre] recites “A method for controlling a predetermined function of a second work process during a first work process in a multitasking mobile terminal that performs the first work process and the second work process.”** This element is met by Hotelling.

152. **Hotelling discloses a method for controlling a predetermined function of a second work process during a first work process in a multitasking mobile terminal that performs the first work process and the second work process** (*e.g.*, method for controlling “Cell Phone” function while “simultaneously” running “Music Player” in a “cell phone”). See, *e.g.*, Hotelling-Prov (emphasis added):

- Hotelling-Prov, Fig. 2

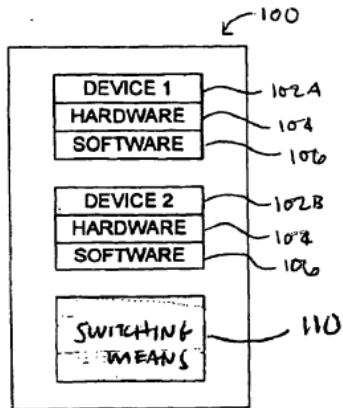


FIG. 2

- Hotelling-Prov, Figs. 10–11

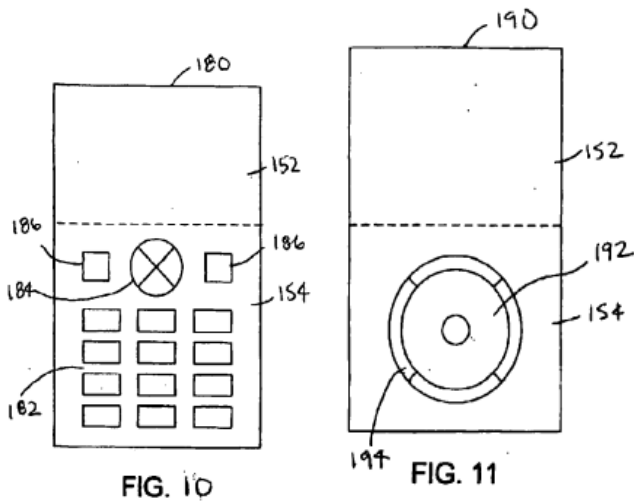


FIG. 10

FIG. 11

- Hotelling-Prov, Fig. 18

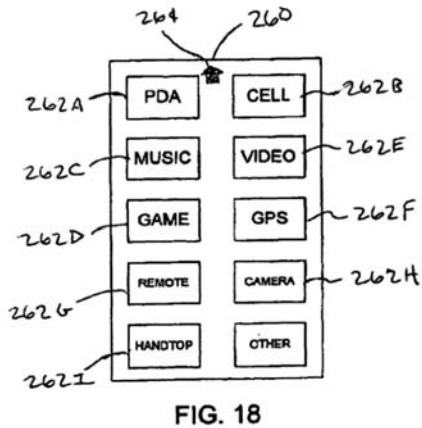


FIG. 18

- **Hotelling-Prov, 7:25-29** (“...*The multi-functional device may, for example, include at least two or more of the following device functionalities: PDA, Cell Phone, Music Player, Video Player, Game Player, Camera, Handtop, Internet terminal, or remote control.*”);
- **Hotelling-Prov., 18:19-29** (“Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the **GUI 260 includes icons/buttons 262 for launching each of the various device functionalities**. In this particular example, the main menu page 260 includes a PDA button 262A, **a cell phone button 262B, a music player button 262C**, a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. **The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260**, and thereafter selects the desired device in the main menu page 260.”);
- **Hotelling-Prov. 20:9-21 (“C. OPERATING AT LEAST TWO FUNCTIONALITIES SIMULTANEOUSLY**
In one embodiment, *the device instead of activating one device, may also activate two devices (or more)*. In cases such as these, *the software for both devices is activated and the display operates in a split screen mode where the screen is parsed into different sections, each section including the device GUI*. By way of example, in a landscape mode, *a first device GUI may be presented on the left side of the display screen* and a second device GUI may be presented on the right side of the display screen. In cases such as this, the GUI would be reduced to accommodate the new space. The dual mode may be selected in a variety of ways. In one implementation, *when the user simultaneously touches two device icons, the device activates both device functionalities and brings both GUIs into view on the screen.*”).
- **See also Hotelling-Prov. 9:12-23, 16:28-31, 16:32-17:5, 20:25-30.**

153. Hotelling discloses a “multi-functional device . . . that has the capabilities of two or more devices in a single device.” Hotelling-Prov., 7:25-29.

The device is a hand-held device that “simultaneously” operates multiple different

functionalities, including for example, “Cell Phone, Music Player, Video Player, Game Player, Camera, Handtop, Internet terminal, or remote control” functionalities. Hotelling-Prov., 7:19-29, 20:9-21. The device stores software for these functions to activate and provide “GUIs” for these functions to a display. Hotelling-Prov., 20:9-21.

(b) Element [1.a]

154. **Element [1.a] recites “displaying one icon in a designated portion of a display, wherein the displayed icon corresponds to the first work process.”** This element is met by Hotelling.

155. **Hotelling discloses displaying one icon in a designated portion of a display (e.g., set location on main menu), wherein the displayed icon corresponds to the first work process (e.g., “icon[]/button[]” for a “music player;” Fig. 18).** See, e.g., Hotelling-Prov (emphasis added):

• **Hotelling-Prov, Fig. 18**

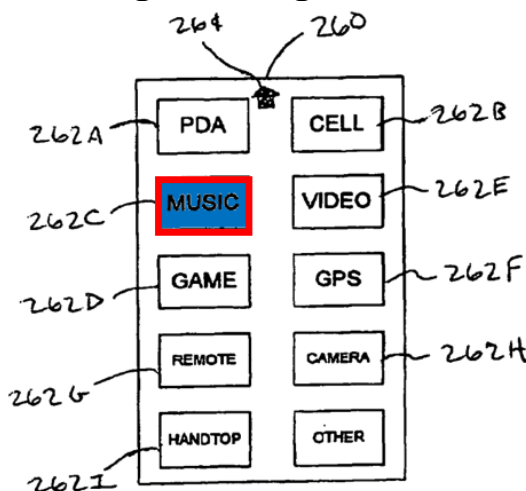


FIG. 18

- **Hotelling-Prov, 18:19-29** (“Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the **GUI 260 includes icons/buttons 262 for launching each of the various device functionalities**. In this particular example, the main menu page 260 includes a PDA button 262A, **a cell phone button 262B, a music player button 262C**, a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. **The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260**, and thereafter selects the desired device in the main menu page 260.”);
- **Hotelling-Prov. 18:31-19:7** (“**The selection may also come in a more active form including flipping (or scrolling) through the various GUIs until the desired GUI is found**. For example, the different GUIs may be incrementally brought into view page after page (or frame after frame) when a next (flip) command signal is generated (e.g., slide show effect). The transition between pages may be widely varied. The transition may be from side to side, top to bottom or center to center. The transition may also include fading in and out, popping in and out, or enlarging and reducing. The command signal may be generated by a physical or virtual button or wheel. Using a button, each press may cause a new page to be displayed. Using a wheel, a predetermined amount of rotation may cause a new page to be displayed.”).
- **See also Hotelling-Prov, 20:25-30.**

156. Hotelling discloses a “main menu page” displayed in the GUI that “includes icons/buttons” corresponding to “various device functionalities.” Hotelling-Prov., 18:19-29. A “music player” icon is displayed in a designated location at “button 262C,” shown in Figure 18. Hotelling-Prov., 18:19-29. A person of ordinary skill in the art would have understood that the icon positions, such as “button 262C,” are fixed on GUI 260 because Hotelling’s icons appear on

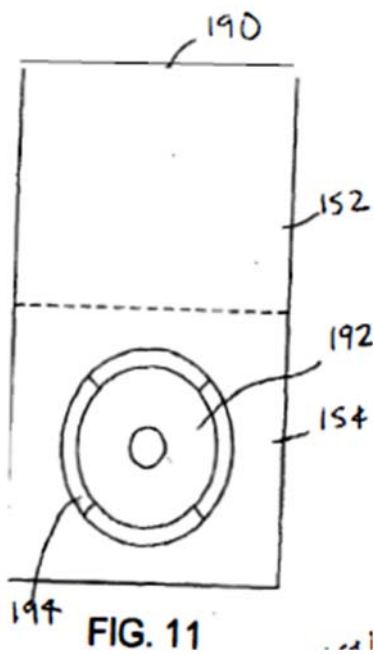
the home page in specified locations, each of which is a designated location.
Hotelling-Prov., 18:19-29, 20:25-30.

(c) Element [1.b]

157. **Element [1.b] recites “if the displayed icon corresponding to the first work process is selected, displaying a screen of the first work process in the display.”** This element is met by Hotelling.

158. **Hotelling discloses, if the displayed icon corresponding to the first work process is selected (see [1.a]; “music player button” is “pressed”), displaying a screen of the first work process in the display (e.g., “the main page for the selected icon...is brought into view on the display”).** See, e.g., Hotelling-Prov. (emphasis added):

- **Hotelling-Prov., Fig. 11**



- **Hotelling-Prov., 16:32-17:5** (“*Fig. 11 is a diagram of a GUI 190 that is used in a Music Player mode. As shown, the GUI 190 is divided into a standard region 152 and a control region 154. Located inside the control region 154 are a virtual scroll wheel 192 and five virtual buttons 194. Additional details on a virtual scroll wheel are provided in U.S. Patent Application No.: 11/038,590, titled, "MODE-BASED GRAPHICAL USER INTERFACES FOR TOUCH SENSITIVE INPUT DEVICES" filed on January 18, 2005.*”);
- **Hotelling-Prov., 18:19-29** (“*Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the GUI 260 includes icons/buttons 262 for launching each of the various device functionalities. In this particular example, the main menu page 260 includes a PDA button 262A, a cell phone button 262B, a music player button 262C, a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260, and thereafter selects the desired device in the main menu page 260.*”).

159. Hotelling discloses that pressing and selecting a “music player button” activates the device’s “Music Player mode” and brings that functionality’s “main page . . . into view on the display.” Hotelling-Prov., 16:32-17:5, 18:19-29, Fig. 11.

(d) Element [1.c]

160. Element [1.c] recites “in response to a first command for sequentially displaying icons, performing sequential display of icons such that an icon corresponding to the second work process is displayed, wherein the icon corresponding to the second work process is displayed in the designated

portion of the display.” This element is met by Hotelling, at least under Patent Owner’s interpretation of the claim.

161. Under Patent Owner’s interpretation, **Hotelling discloses in response to a first command for sequentially displaying icons, performing sequential display of icons** (e.g., selecting a “home button” causes “device” to “load the main menu page”) **such that an icon corresponding to the second work process is displayed, wherein the icon corresponding to the second work process is displayed in the designated portion of the display** (e.g., “cell phone” icon corresponding to “cell phone” application is displayed in its designated location 262B on the “main menu”). *See, e.g., Hotelling-Prov. (emphasis added):*

- **Hotelling-Prov., Fig. 18**

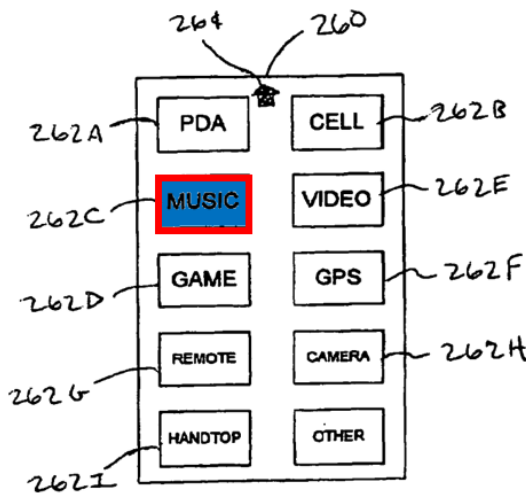


FIG. 18

- **Hotelling-Prov., 18:19-29** (“Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the **GUI 260 includes icons/buttons 262 for launching each of the various device functionalities**. In this particular example, the main menu page 260 includes a PDA button 262A, **a cell phone button 262B, a music player button**

262C, a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. ***The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260, and thereafter selects the desired device in the main menu page 260.***)

162. See my discussion of element [1.a]. As explained for element [1.a] above, Hotelling discloses displaying work process icons in a “designated portion of the display.” See [1.a]. Further, I understand that Patent Owner has taken the position that “swipe up” and “press home button” commands that cause a display to present a screen with an icon in a designated location meet the claimed command for sequentially displaying icons. Ex. 1031, 9–10 (contending that “swipe up” command is a command for sequentially displaying icons), 12 (contending that “press home button” is a command for sequentially displaying icons). Hotelling discloses this element under the Patent Owner’s interpretation. Hotelling discloses a command for “select[ing] a soft home button 264” to return to “main menu page 260,” which causes a menu of icons to be displayed. Hotelling-Prov. 18:19-29. The icons include a “cell phone” function icon, which, as shown in Figure 18, is located in a designated portion at “button 262B” on the GUI. Hotelling-Prov., 18:19-20, Fig. 18.

(e) Element [1.d]

163. Element [1.d] recites “in response to a second command for changing to the second work process, replacing the screen of the first work process with a screen of the second work process.” This element is met by Hotelling.

164. Hotelling discloses in response to a second command for changing to the second work process, replacing the screen of the first work process with a screen of the second work process (e.g., “When the [cell phone] button is pressed, the main page for the selected icon...is brought into view”). See, e.g., Hotelling-Prov. (emphasis added):

- Hotelling-Prov., Fig. 10

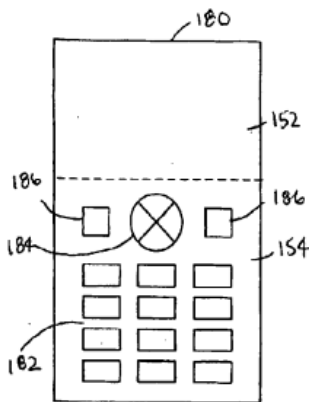


FIG. 10

- Hotelling-Prov. 16:28-31 (“Fig. 10 is a diagram of a GUI 180 that is used in a Cell Phone mode. As shown, the GUI 180 is divided into a standard region 152 and a control region 154. Located inside the control region 154 are a virtual keypad 182, a virtual navigation pad 184 and two virtual buttons 186.”);
- Hotelling-Prov., 18:19-29 (“Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the GUI 260

includes icons/buttons 262 for launching each of the various device functionalities. In this particular example, the main menu page 260 includes a PDA button 262A, *a cell phone button 262B, a music player button 262C*, a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. *The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260*, and thereafter selects the desired device in the main menu page 260.”).

- **See also Hotelling-Prov, 18:31-19:7.**

165. See my discussion of element [1.b]. As explained for element [1.b] above, Hotelling discloses if an icon corresponding to a work process is selected, displaying a screen of the work process. See [1.b]. Hotelling further discloses that pressing and selecting a “cell phone button” activates the device’s “Cell Phone mode” and brings the cell phone interface onto the display. Hotelling-Prov., 16:28-31, 18:19-29, Fig. 10.

(f) Element [1.e]

166. **Element [1.e] recites “controlling the multitasking of the second work process and the first work process.”** This element is met by Hotelling.

167. **Hotelling discloses controlling the multitasking of the second work process and the first work process (e.g., different functionalities, including “Cell Phone” and “Music Player” functionalities, are operated “simultaneously”).** See, e.g., Hotelling-Prov. (emphasis added):

- **Hotelling-Prov, 7:25-29** (“...*The multi-functional device may, for example, include at least two or more of the following device functionalities: PDA, Cell Phone, Music Player, Video Player, Game Player, Camera, Handtop, Internet terminal, or remote control.*”);
- **Hotelling-Prov. 20:9-21** (“**C. OPERATING AT LEAST TWO FUNCTIONALITIES SIMULTANEOUSLY**
In one embodiment, *the device instead of activating one device, may also activate two devices (or more).* In cases such as these, *the software for both devices is activated and the display operates in a split screen mode where the screen is parsed into different sections, each section including the device GUI.* By way of example, in a landscape mode, *a first device GUI may be presented on the left side of the display screen and a second device GUI may be presented on the right side of the display screen.* In cases such as this, the GUI would be reduced to accommodate the new space. The dual mode may be selected in a variety of ways. In one implementation, *when the user simultaneously touches two device icons, the device activates both device functionalities and brings both GUIs into view on the screen.*”).
- **See also Hotelling-Prov., 18:19-29.**

168. See my discussion of element [1.pre]. As explained for element [1.pre] above, Hotelling discloses a first work process and a second work process in a multitasking device. See [1.pre]. Hotelling further teaches its functionalities are operated “simultaneously,” for example, so that the “Music Player” plays music while the “Cell Phone” GUI number pad is displayed. Hotelling-Prov., 7:25-29, 20:9-21.

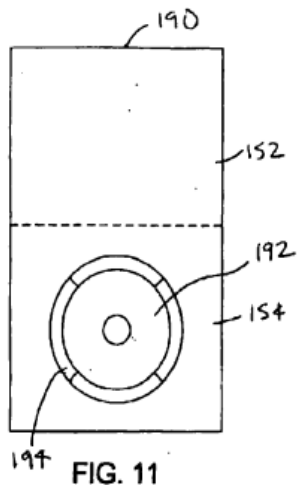
(g) Element [1.f]

169. Element [1.f] recites “in response to a third command for activating a control menu for controlling a predetermined function of the first

work process, activating the control menu for controlling the predetermined function of the first work process.” This element is met by Hotelling.

170. **Hotelling discloses in response to a third command for activating a control menu for controlling a predetermined function of the first work process (e.g., simultaneously touching “music player” and “Cell Phone” to display “split screen mode”; activate “floating controls” with “music player” “play, pause” buttons; “invoke the virtual control knob” to control “volume” of a “currently playing” “song”), activating the control menu for controlling the predetermined function of the first work process (e.g., “music player” in “split screen mode” includes playback controls; “floating controls” include “play, pause” “control buttons”; knob controls “volume” function for a “currently playing” “song”).** See, e.g., Hotelling-Prov. (emphasis added):

- **Hotelling-Prov., Fig. 11**



- **Hotelling-Prov., Fig. 18**

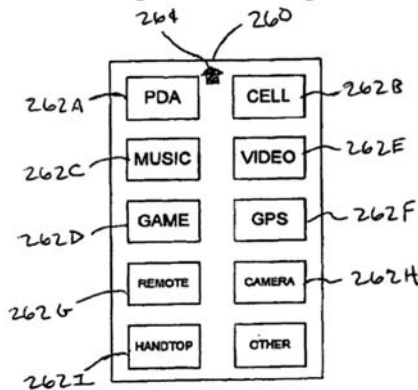


FIG. 18

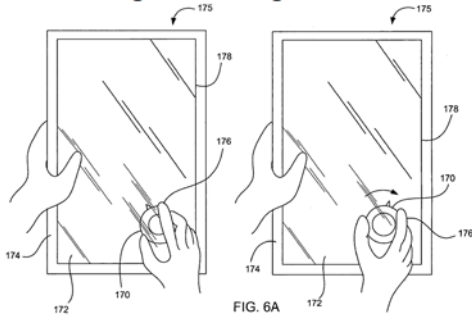
- **Hotelling-Prov. 8:29-9:2** (“Music players are generally configured to store, process and output music. ***Music players can be based on the MP3 or AAC format***, which is a compression system for music. ***Music Players typically include a microprocessor, memory, display, audio jack, data port and playback controls. The playback controls typically include features such as menu, play/pause, next, previous, volume up, and volume down.***”);
- **Hotelling-Prov. 15:1-15** (“In some cases, the entire display is used to present this information, and in other cases only a portion of the display may be used for the GUI controls. In the later case, for example, and referring to Figs. 5-6, the GUI 150 may be separated into 5 a standard region 152 and a control region 154. The standard region 152 represents what would normally be displayed on the display 122 when using a particular device. That is, the standard GUI screens associated with the selected device are displayed in the standard region. For example, in the case of the PDA, a main menu (window with a set of icons), calendar, address book or date book may be displayed in the standard 10 region 152. ***The control region 154, on the other hand, virtually represents the physical controls that would normally be physically placed on a particular device.*** That is, the virtual controls that mimic the physical controls are displayed in the control region 154. For example, in the case of the PDA, the control region 154 may include virtual representations of a hand writing recognition area, a navigation pad and the standard function buttons.”);
- **Hotelling-Prov., 16:32-17:5** (“***Fig. 11 is a diagram of a GUI 190 that is used in a Music Player mode. As shown, the GUI 190 is divided into a standard region 152 and a control region 154.*** Located inside the control region 154 are a virtual scroll wheel 192 and five virtual buttons 194.

Additional details on a virtual scroll wheel are provided in U.S. Patent Application No.: 11/038,590, titled, "MODE-BASED GRAPHICAL USER INTERFACES FOR TOUCH SENSITIVE INPUT DEVICES" filed on January 18, 2005.”);

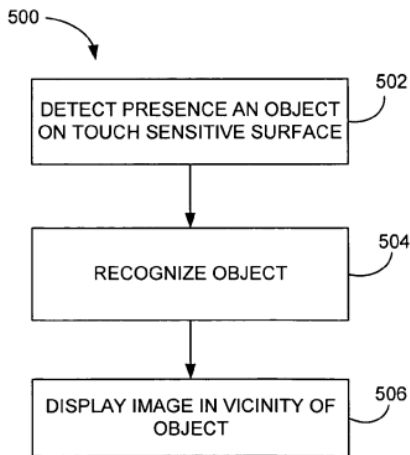
- **Hotelling-Prov. 18:31-19:7** (“*The selection may also come in a more active form including flipping (or scrolling) through the various GUIs until the desired GUI is found.* For example, the different GUIs may be incrementally brought into view page after page (or frame after frame) when a next (flip) command signal is generated (e.g., slide show effect). The transition between pages may be widely varied. The transition may be from side to side, top to bottom or center to center. The transition may also include fading in and out, popping in and out, or enlarging and reducing. The command signal may be generated by a physical or virtual button or wheel. Using a button, each press may cause a new page to be displayed. Using a wheel, a predetermined amount of rotation may cause a new page to be displayed.”);
- **Hotelling-Prov. 20:9-21** (“**C. OPERATING AT LEAST TWO FUNCTIONALITIES SIMULTANEOUSLY**
In one embodiment, *the device instead of activating one device, may also activate two devices (or more).* In cases such as these, *the software for both devices is activated and the display operates in a split screen mode where the screen is parsed into different sections, each section including the device GUI.* By way of example, in a landscape mode, *a first device GUI may be presented on the left side of the display* screen and a second device GUI may be presented on the right side of the display screen. In cases such as this, the GUI would be reduced to accommodate the new space. The dual mode may be selected in a variety of ways. In one implementation, *when the user simultaneously touches two device icons, the device activates both device functionalities and brings both GUIs into view on the screen.*”);
- **Hotelling-Prov. 33:5-12** (“*In another embodiment, the hand-held device may include a navigation pad.* The navigation pad is typically included with many hand-held devices. The only problem with including a navigation pad is that it takes up a large amount of real estate. In some cases, the functionality of the navigation pad changes according to the mode of the device. For example, the directional keys may be assigned different functionality. *In the case of a music player for example, the directional keys may be assigned, play/pause, Next, Previous, and volume up and*

down. Other assignable buttons may also be included on the device (although in a limited number).”).

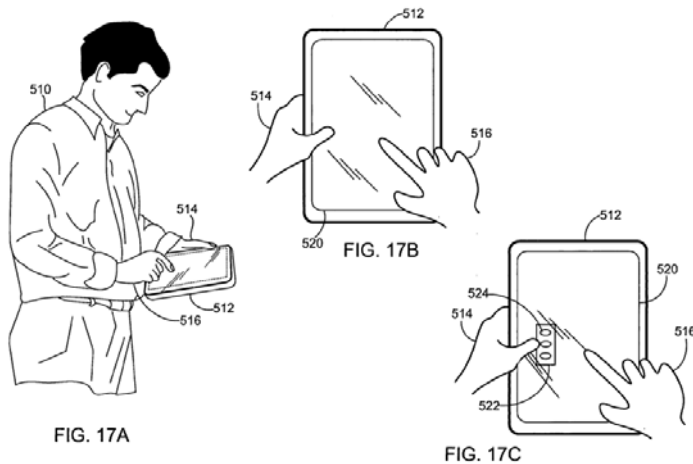
- **See also Hotelling-Prov., 18:19-29.**
- **Hotelling-521, Fig. 6A**



- **Hotelling-521, Fig. 16**



- **Hotelling-521, Figs. 17A-17C**



- **Hotelling-521 [0086]** (“To cite an example using the above methodologies, and referring to FIGS. 6A-6H, *a rotate gesture for controlling a virtual volume knob 170 on a GUI interface 172 of a display 174 of a tablet PC 175 will be described. In order to actuate the knob 170, the user places their fingers 176 on a multipoint touch screen 178.* The virtual control knob may already be displayed, or *the particular number, orientation or profile of the fingers at set down, or the movement of the fingers immediately thereafter, or some combination of these and other characteristics of the user's interaction may invoke the virtual control knob to be displayed.* In either case, the computing system associates a finger group to the virtual control knob and makes a determination that the user intends to use the virtual volume knob. This association may also be based in part on the mode or current state of the computing device at the time of the input. For example, *the same gesture may be interpreted alternatively as a volume knob gesture if a song is currently playing on the computing device, or as a rotate command if an object editing application is being executed.* Other user feedback may be provided, including for example audible or tactile feedback.”);
- **Hotelling-521 [0093]** (“As shown in FIG. 6G, *based on the rotation vector the speaker 192 of the tablet PC 175 increases (or decreases) its output in accordance with the amount of rotation of the fingers 176, i.e., increase the volume by 5% based on rotation of 5 degrees.* The display 174 of the tablet PC can also adjust the rotation of the knob 170 in accordance with the amount of rotation of the fingers 176, i.e., the position of the knob 170 rotates five (5) degrees. In most cases, the rotation of the knob occurs simultaneously with the rotation of the fingers, i.e., for every degree of

finger rotation the knob rotates a degree. In essence, the virtual control knob follows the gesture occurring on the screen. Still further, an audio unit 194 of the tablet PC may provide a clicking sound for each unit of rotation, e.g., provide five clicks based on rotation of five degrees. Still yet further, a haptics unit 196 of the tablet PC 175 may provide a certain amount of vibration or other tactile feedback for each click thereby simulating an actual knob.”);

- **Hotelling-521 [0111]** (“FIG. 16 is a diagram of a GUI operational method 500, in accordance with one embodiment of the present invention. ***The GUI operational method 500 is configured for initiating floating controls in a GUI. The GUI operational method 500 generally begins at block 502 where the presence of an object such as a finger or thumb is detected.*** This may for example be accomplished using a touch screen. Following block 502, the GUI operational method 500 proceeds to block 504 where the object is recognized (the identity of the object is found). The object may be recognized among a plurality of objects. For example, see block 104 of FIG. 2 above.”);
- **Hotelling-521 [0112]** (“***Following block 504, the GUI operational method 500 proceeds to block 506 where an image in the vicinity of the object is generated.*** The image is typically based on the recognized object. The image may include windows, fields, dialog boxes, menus, icons, buttons, cursors, scroll bars, etc. In some cases, the user can select and activate the image (or features embedded therein) in order to initiate functions and tasks. By way of example, the image may be a user interface element or a group of user interface elements (e.g., one or more buttons that open, close, minimize, or maximize a window). The image may also be one or more icons that launch a particular program or files that open when selected. The image may additionally correspond to non interactive text and graphics. In most cases, the image is displayed as long as the object is detected or it may be displayed for some preset amount of time, i.e., after a period of time it times out and is removed.);
- **Hotelling-521 [0113]** (“In one particular embodiment, ***the image includes one or more control options that can be selected by the user.*** The control options may include one or more control buttons for implementing various tasks. ***For example, the control option box may include music listening control buttons as for example, play, pause, seek and menu.***”);
- **See also Hotelling-521 Figs. 6B-6G; [0085], [0087]-[0092], [0094], [0114].**

171. Hotelling teaches at least three different features that disclose the claimed activated control menu. First, Hotelling discloses a “split screen” mode in which the “music player” and “Cell Phone” GUIs are displayed side by side. Hotelling-Prov., 8:29-9:2; 15:1-15; 16:32-16:5, 18:19-29, 20:9-21. The user activates the split screen mode, for example, by “simultaneously touch[ing] two device icons.” Hotelling-Prov. 20:9-21. In a split screen where the “music player” is displayed side by side with the “cell phone,” the “music player” GUI includes “playback controls” such as “play/pause, next, previous, volume up, and volume down” listed in menus that include controls for operating the application. Hotelling-Prov., 8:29-9:2. Alternatively, to the extent the claimed control menu must be displayed with a screen of the second work process, the cell phone function, Hotelling’s “split-screen” embodiment in which the Music Player “control region” is displayed with the “cell phone” GUI discloses this element. Hotelling-Prov., 16:28-31 (“Fig. 10 is a diagram of a GUI 180 that is used in a Cell Phone mode.”); 16:32-17:5 (“Fig. 11 is a diagram of a GUI 190 that is used in a Music Player mode.”); 20:9-21 (“... software for both devices is activated and the display operates in a split screen mode”).

172. Second, Hotelling-521 discloses touching the device screen to activate “floating controls.” Hotelling-521 [0111]. I understand that because Hotelling explicitly incorporates Hotelling-521 by reference, the disclosures of Hotelling-521

are effectively incorporated in Hotelling and, for the purposes of prior art disclosure, are treated as if they are part of Hotelling's own disclosures. The user activates floating controls by touching the device so that the device "detect[s]" "the presence of an object such as a finger or thumb." Hotelling-521 [0111]-[0112]. The floating controls include "control buttons . . . for example, play pause, seek and menu," each of which controls functions of the music player application—e.g., by playing, pausing, skipping through songs, navigating the music application, and so on Hotelling-521 [0113].

173. Third, Hotelling-521 also discloses using a gesture to "invoke [a] virtual control knob." Hotelling-521 [0086]. The control knob is used for controlling "volume" of "currently playing" music—indicating that the "volume" being controlled by this menu is specifically for the particular application. Hotelling-521 [0086]. The control knob is a "volume" control menu, according to Patent Owner's own statements about what constitutes a control menu for the '631 Patent. Hotelling-521 [0086]. See Ex. 1031, 21. Indeed, the '631 patent itself describes "volume control" constitutes a function of an MP3 playback process. '631 patent, 2:40-44.

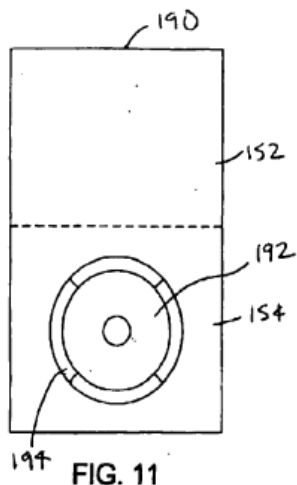
(h) Element [1.g]

174. **Element [1.g] recites "in response to a fourth command for controlling the predetermined function of the first work process through the**

activated control menu, controlling the predetermined function of the first work process through the activated control menu.” This element is met by Hotelling.

175. Hotelling discloses in response to a fourth command for controlling the predetermined function of the first work process through the activated control menu, controlling the predetermined function of the first work process through the activated control menu (see [1.f], e.g., “Music Player” “play/pause” controls in “split screen” control a playback function; “floating controls” include “play” and “pause” buttons to control a playback function; “volume control knob” controls “volume” functions). See, e.g., Hotelling-Prov. (emphasis added):

- Hotelling-Prov., Fig. 11



- Hotelling-Prov., 8:29-9:2 (“Music players are generally configured to store, process and output music. **Music players can be based on the MP3 or AAC format**, which is a compression system for music. **Music Players typically**

include a microprocessor, memory, display, audio jack, data port and playback controls. The playback controls typically include features such as menu, play/pause, next, previous, volume up, and volume down.”);

- **Hotelling-Prov. 15:1-15** (“In some cases, the entire display is used to present this information, and in other cases only a portion of the display may be used for the GUI controls. In the later case, for example, and referring to Figs. 5-6, the GUI 150 may be separated into a standard region 152 and a control region 154. The standard region 152 represents what would normally be displayed on the display 122 when using a particular device. That is, the standard GUI screens associated with the selected device are displayed in the standard region. For example, in the case of the PDA, a main menu (window with a set of icons), calendar, address book or date book may be displayed in the standard region 152. ***The control region 154, on the other hand, virtually represents the physical controls that would normally be physically placed on a particular device.*** That is, the virtual controls that mimic the physical controls are displayed in the control region 154. For example, in the case of the PDA, the control region 154 may include virtual representations of a hand writing recognition area, a navigation pad and the standard function buttons.”);
- **Hotelling-Prov., 16:32-17:5** (“***Fig. 11 is a diagram of a GUI 190 that is used in a Music Player mode. As shown, the GUI 190 is divided into a standard region 152 and a control region 154.*** Located inside the control region 154 are a virtual scroll wheel 192 and five virtual buttons 194. Additional details on a virtual scroll wheel are provided in U.S. Patent Application No.: 11/038,590, titled, "MODE-BASED GRAPHICAL USER INTERFACES FOR TOUCH SENSITIVE INPUT DEVICES" filed on January 18, 2005.”);
- **Hotelling-Prov. 20:9-21** (“***C. OPERATING AT LEAST TWO FUNCTIONALITIES SIMULTANEOUSLY***
In one embodiment, ***the device instead of activating one device, may also activate two devices (or more).*** In cases such as these, ***the software for both devices is activated and the display operates in a split screen mode where the screen is parsed into different sections, each section including the device GUI.*** By way of example, in a landscape mode, ***a first device GUI may be presented on the left side of the display*** screen and a second device GUI may be presented on the right side of the display screen. In cases such as this, the GUI would be reduced to accommodate the new space. The dual

mode may be selected in a variety of ways. In one implementation, ***when the user simultaneously touches two device icons, the device activates both device functionalities and brings both GUIs into view on the screen.***”).

- **Hotelling-521, [0093]** (“As shown in FIG. 6G, ***based on the rotation vector the speaker 192 of the tablet PC 175 increases (or decreases) its output in accordance with the amount of rotation of the fingers 176, i.e., increase the volume by 5% based on rotation of 5 degrees.*** The display 174 of the tablet PC can also adjust the rotation of the knob 170 in accordance with the amount of rotation of the fingers 176, i.e., the position of the knob 170 rotates five (5) degrees. In most cases, the rotation of the knob occurs simultaneously with the rotation of the fingers, i.e., for every degree of finger rotation the knob rotates a degree. In essence, the virtual control knob follows the gesture occurring on the screen. Still further, an audio unit 194 of the tablet PC may provide a clicking sound for each unit of rotation, e.g., provide five clicks based on rotation of five degrees. Still yet further, a haptics unit 196 of the tablet PC 175 may provide a certain amount of vibration or other tactile feedback for each click thereby simulating an actual knob.”);
- **Hotelling-521, [0111]** (“FIG. 16 is a diagram of a GUI operational method 500, in accordance with one embodiment of the present invention. ***The GUI operational method 500 is configured for initiating floating controls in a GUI. The GUI operational method 500 generally begins at block 502 where the presence of an object such as a finger or thumb is detected.*** This may for example be accomplished using a touch screen. Following block 502, the GUI operational method 500 proceeds to block 504 where the object is recognized (the identity of the object is found). The object may be recognized among a plurality of objects. For example, see block 104 of FIG. 2 above.”);
- **Hotelling-521 [0113]** (“In one particular embodiment, ***the image includes one or more control options that can be selected by the user.*** The control options may include one or more control buttons for implementing various tasks. ***For example, the control option box may include music listening control buttons as for example, play, pause, seek and menu.***”).
- **See also Hotelling-521 [0090], [0114].**

176. See my discussion of element [1.f]. As explained for element [1.f]

above, Hotelling discloses an activated control menu for controlling a

predetermined function of the first work process. See [1.f]. In the “split screen” feature described for [1.f], Hotelling discloses the “control region” includes “play/pause” buttons that a user interacts with to control music playback functions. Hotelling-Prov., 8:29-9:2, 16:32-17:5, 20:9-21. In the “floating controls” feature described for [1.f], Hotelling-521 also discloses play/pause buttons that a user interacts with to control playback functions such as play and pause functions. Hotelling-521 [0113]. In the “control knob” feature discussed for [1.f], Hotelling-521 discloses a knob that a user turns to control the volume of music playback. Hotelling-521 [0093].

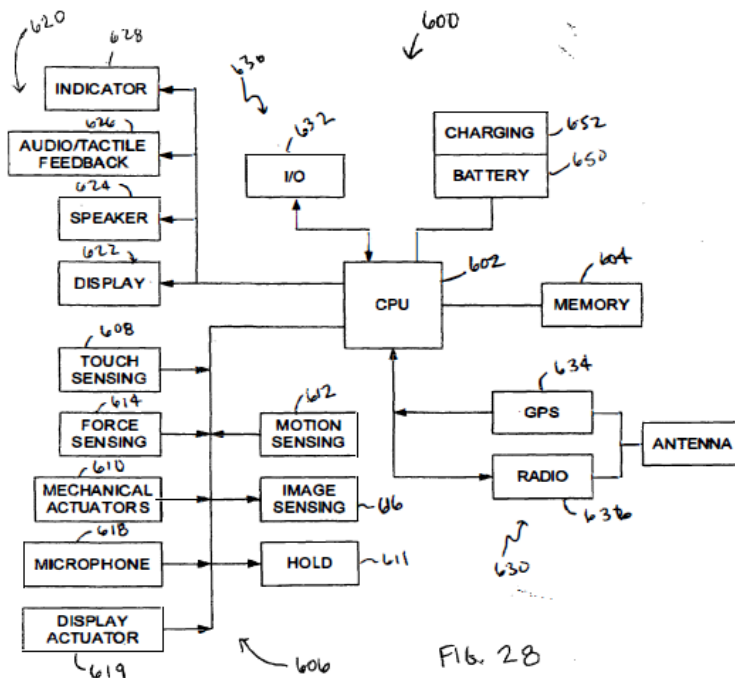
(i) Claim [2]

177. **Claim [2] recites “The method of claim 1, wherein the controlling the predetermined function of the first work process comprises: generating a control signal for controlling the predetermined function of the first work process in response to the fourth command; and controlling the predetermined function of the first work process in response to the generated control signal.”** This Claim is met by Hotelling.

178. **Hotelling further discloses controlling the predetermined function of the first work process comprises: generating a control signal for controlling the predetermined function of the first work process in response to the fourth command** (*e.g.*, see [1.f]-[1.g]; “controller” “interprets” a user “touch” as “input

signals” for “Music Player” functions (see [1.g])); **and controlling the predetermined function of the first work process in response to the generated control signal** (e.g., see [1.f]-[1.g]; “controller” will control a playback functions, including play, pause, and volume, in response to the “input signals”). See, e.g., Hotelling-Prov. (emphasis added):

- **Hotelling-Prov., Fig. 28**



- **Hotelling-Prov. 8:29-9:2** (“Music players are generally configured to store, process and output music. **Music players can be based on the MP3 or AAC format**, which is a compression system for music. **Music Players typically include a microprocessor, memory, display, audio jack, data port and playback controls. The playback controls typically include features such as menu, play/pause, next, previous, volume up, and volume down.**”);
- **Hotelling-Prov., 22:6-20** (“In accordance with one embodiment of the present invention, a touch screen is configured as the primary input means of the hand-held device (in conjunction with GUI). The touch screen is a transparent input panel positioned in front of the display. **The touch screen**

generates input signals when an object such as a finger (or stylus) is moved across the surface of the touch screen, from an object holding a particular position on the touch screen and/or by a finger tapping on the touch screen. In most cases, touch screens allow a user to make selections and initiate movements in a GUI by simply touching the display screen via a finger. For example, a user may make a selection by pointing directly to a graphical object displayed on the display screen. 1 5 The graphical object may, for example, correspond to an on-screen button for performing specific actions in the hand-held electronic device. ***In general, the touch screen recognizes the touch and position of the touch on the display and a controller of the hand-held electronic device interprets the touch and thereafter performs an action based on the touch event.*** There are several types of touch screen technologies 20 including resistive, capacitive, infrared and surface acoustic wave.”);

- **Hotelling-Prov., 47:31-48:10** (“10. OVERALL BLOCK DIAGRAM Fig. 28 is a block diagram of an exemplary hand-held device 600 in accordance with one embodiment of the present invention. ***The hand-held device 600 typically includes a controller 602 (e.g., CPU) configured to execute instructions and to carry out operations associated with the hand-held device.*** For example, using instructions retrieved for example from memory, the ***controller 602 may control the reception and manipulation of input and output data between components of the hand-held device 600.*** The controller 602 can be implemented on a single-chip, multiple chips or multiple electrical components. For example, various architectures can be used for the controller 602, including dedicated or embedded processor, single purpose processor, controller, ASIC, and so forth. By way of example, the controller may include microprocessors, DSP, A/D converters, D/A converters, compression, 10 decompression, etc.”).
- **See also Hotelling-Prov. 48:12-26, 48:28-49:3, 48:5-9, 49:11-15.**

179. See my discussion of elements [1.pre], [1.f]–[1.g]. As explained for [1.pre] and [1.f]–[1.g] above, Hotelling discloses controlling a predetermined function of a first work process in response to a fourth command. See [1.pre], [1.f]–[1.g]. Hotelling further discloses the mobile device includes “controller 602”

that “execute[s] instructions” and “carr[ies] out operations” associated with the device. Hotelling-Prov., 47:31-48:10. The controller controls the “reception and manipulation of input and output data between components of the hand-held device.” Hotelling-Prov., 47:31-48:10. The touch screen display “generates input signals when an object such as a finger (or stylus) is moved across the surface” and the controller receives the “input signal.” Hotelling-Prov., 22:6-20. Thus, when interacting with a touch screen, this generates an input signal that is received by the control and is used to execute instructions and operate the device via the controller. This operation of the input signal therefore not only renders obvious, but affirmatively discloses that when a command is input as an input signal by a user, this causes the controller to generate a corresponding control signal that is used to execute instructions carrying out the requested action. For example, an input signal from the user manipulating controls of the “music player” application will result in control signals instructing the “Music Player” application play, pause, or control the volume of music playing on the device. Hotelling-Prov., 8:29-9:2, 22:6-20.

(j) Claim [3]

180. **Claim [3] recites “The method of claim 1, further comprising: in response to a fifth command for deactivating the activated control menu, deactivating the activated control menu.”** This element is met by Hotelling.

181. **Hotelling** discloses in response to a fifth command for deactivating the activated control menu, deactivating the activated control menu (e.g., controls “disappear” “when the thumb is moved away from the touch screen”; floating controls are “closed using conventional close icons or buttons”). See, e.g., Hotelling (emphasis added):

- **Hotelling-521, Fig. 17C**

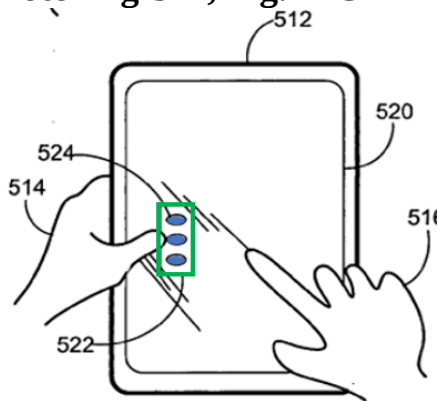


FIG. 17C

- **Hotelling-Prov., Fig. 8**

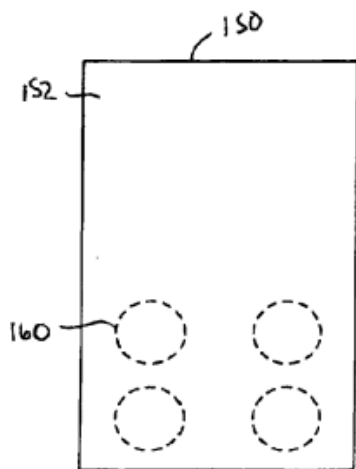


FIG. 8

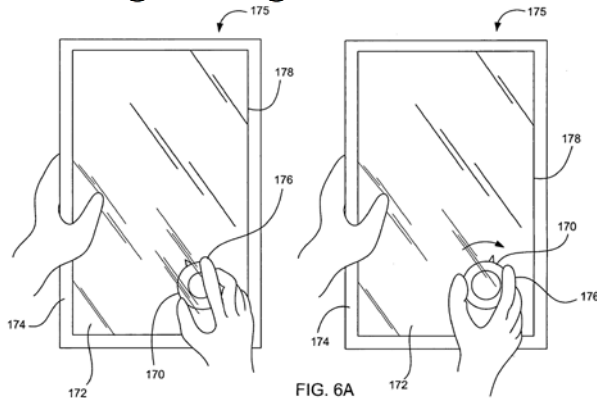
- **Hotelling-Prov., 16:8-16** (“Alternatively, as shown in FIG. 8, *virtual controls 160 may be overlaid on top of the standard region 152 so that the*

standard region 152 can fully utilize the entire display 122. In fact, the virtual controls 160 may appear and disappear as needed. For example, the user may touch the screen and this may drive the device to display the controls over a portion of the display including whatever is already displayed. Examples of virtual controls that operate in this manner can be found in U.S. Patent Application No.: 11/038,590, titled, 'MODE-BASED GRAPHICAL USER INTERFACES FOR TOUCH SENSITIVE INPUT DEVICES' filed on January 18, 2005.”);

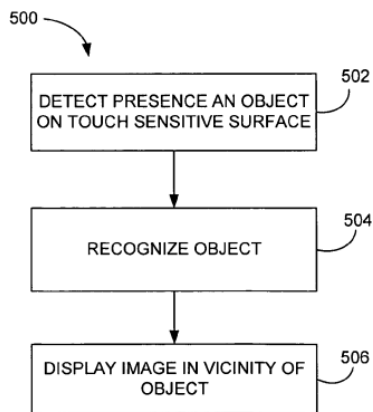
- **Hotelling-Prov., 16:32-17:5** (“***Fig. 11 is a diagram of a GUI 190 that is used in a Music Player mode. As shown, the GUI 190 is divided into a standard region 152 and a control region 154.*** Located inside the control region 154 are a virtual scroll wheel 192 and five virtual buttons 194. Additional details on a virtual scroll wheel are provided in U.S. Patent Application No.: 11/038,590, titled, "MODE-BASED GRAPHICAL USER INTERFACES FOR TOUCH SENSITIVE INPUT DEVICES" filed on January 18, 2005.”);
- **Hotelling-Prov., 18:19-29** (“***Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the GUI 260 includes icons/buttons 262 for launching each of the various device functionalities.*** In this particular example, the main menu page 260 includes a PDA button 262A, ***a cell phone button 262B, a music player button 262C,*** a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. ***The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260,*** and thereafter selects the desired device in the main menu page 260.”);
- **Hotelling-Prov. 18:31-19:7** (“***The selection may also come in a more active form including flipping (or scrolling) through the various GUIs until the desired GUI is found.*** For example, the different GUIs may be incrementally brought into view page after page (or frame after frame) when a next (flip) command signal is generated (e.g., slide show effect). The transition between pages may be widely varied. The transition may be from side to side, top to bottom or center to center. The transition may also include fading in and out, popping in and out, or enlarging and reducing.

The command signal may be generated by a physical or virtual button or wheel. Using a button, each press may cause a new page to be displayed. Using a wheel, a predetermined amount of rotation may cause a new page to be displayed.”);

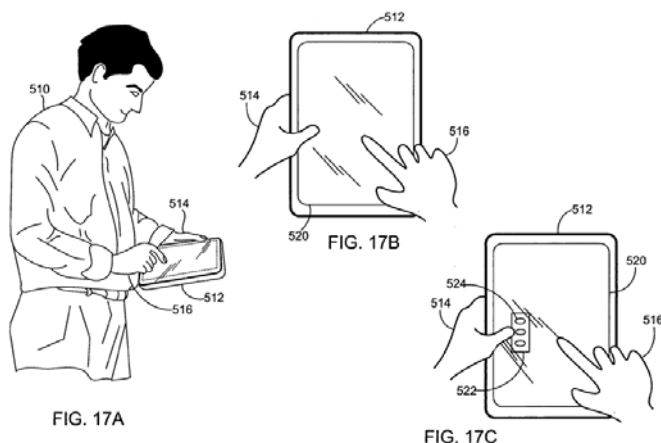
- **Hotelling-521, Fig. 6A**



- **Hotelling-521, Fig. 16**



- **Hotelling-521, Figs. 17A–17C**



- **Hotelling-521, [0065]** (“*When a user performs one or more gestures, the input device 70 relays gesture information to the processor 56. Using instructions from memory 58, and more particularly, the gestural operational program 88, the processor 56 interprets the gestures 85 and controls different components of the computer system 50, such as memory 58, a display 68 and I/O devices 80, based on the gestures 85.* The gestures 85 may be identified as commands for performing actions in applications stored in the memory 58, modifying GUI objects shown on the display 68, modifying data stored in memory 58, and/or for performing actions in I/O devices 80. By way of example, the commands may be associated with zooming, panning, scrolling, paging, rotating, sizing, and the like. As further examples, the commands may also be associated with launching a particular program, opening a file or document, viewing a menu, making a selection, executing instructions, logging onto the computer system, permitting authorized individuals access to restricted areas of the computer system, loading a user profile associated with a user's preferred arrangement of the computer desktop, and/or the like.”);
- **Hotelling-521 [0113]** (“In one particular embodiment, *the image includes one or more control options that can be selected by the user.* The control options may include one or more control buttons for implementing various tasks. *For example, the control option box may include music listening control buttons as for example, play, pause, seek and menu.*”);
- **Hotelling-521 [0114]** (“FIGS. 17A-17E illustrate a floating control sequence using the method described above. As shown in FIG. 17A, a user 510 is using a tablet PC 512 and therefore is holding the tablet PC 512 with one hand 514 while navigating (e.g., tracking, gesturing) with the other hand

516. As shown in FIG. 17B, which is a close up of the user holding the tablet PC 512, a portion of the thumb of the holding hand 514 is positioned over the touch screen 520. As shown in FIG. 17C, the tablet PC 512 recognizes the thumb and displays a control box 522 adjacent the thumb.

The control box 522 includes various buttons 524, which can be selected by the user's thumb to initiate tasks in the tablet PC 512. As shown in FIG. 17D, while holding the tablet PC 512, the thumb is extended over one of the buttons 524 and subsequently tapped thereby selecting the task associated with the button 524. By way of example, the task may be associated with launching a program or gaining access to a network or changing the mode of operation of the device. The control box 522 and buttons 524 may be used to change the input mode of the touch screen 520 so that, for example, the identical gesture made with the fingers of the user's other hand may have multiple meanings depending on which of buttons 524 is selected. ***As shown in FIG. 17E, when the thumb is moved away from the touch screen 520, the control box 522 may time out and disappear. Alternatively, the control box may be closed using conventional close icons or buttons.***”).

- ***See also Hotelling-521 [0111]-[0112].***

182. See my discussion of elements [1.f]–[1.g]. As explained for [1.f]–[1.g] above, Hotelling discloses a third command for activating a control menu. See [1.f]–[1.g]. Hotelling further teaches deactivating the control menu feature discussed for [1.f]. For Hotelling’s “split-screen” feature, the music player’s “control region” may be deactivated by navigating away from the GUI. See Hotelling-Prov., 16:32-17:5, 18:19-19:7, Fig. 18. For example, opening the “main menu page” while the music player’s control region is open will close the music player’s control region, thereby closing that control menu. Hotelling-Prov., 18:19-19:7. For Hotelling-521’s “floating controls” feature, the “floating controls” include “conventional” “icons or buttons” for deactivating the control menu; such

buttons include a close button, which deactivates the menu. Hotelling-521, Fig. 17C, [0065], [0113]–[0114]. For Hotelling-521’s volume “control knob” feature, Hotelling not only renders obvious but affirmatively discloses closing such a menu. It would have been understood that users would need a mechanism for closing such a menu after it has been opened for the device to operate effectively. *See* Hotelling-521 [0114]. Hotelling-521 itself teaches commands closing menus in a variety of ways, including “conventional close icons,” gesturing a thumb or finger “away from the touch screen,” and simply allowing the controls to “time out and disappear.” Hotelling-521 [0085]–[0086], [0113]–[0114]. Each of these mechanisms for closing a menu applies to the volume control menu. Hotelling therefore not only renders obvious but affirmatively discloses that such a menu would be closed using one of the disclosed techniques, including conventional techniques, for closing menus. Hotelling-521 [0085]–[0086], [0113]–[0114].

(k) Claim [4]

183. **Claim [4] recites “The method of claim 1, wherein the first work process is an MP3 play process.”** This element is met by Hotelling.

184. **Hotelling further discloses wherein the first work process is an MP3 play process** (*e.g.*, the first work process (*see* [1.a]) plays “music” based on an “MP3” format). *See, e.g.*, Hotelling-Prov. (emphasis added):

- **Hotelling-Prov. 8:29-9:2** (“Music players are generally configured to store, process and output music. *Music players can be based on the MP3 or AAC format*, which is a compression system for music. *Music Players typically include a microprocessor, memory, display, audio jack, data port and playback controls. The playback controls typically include features such as menu, play/pause, next, previous, volume up, and volume down.*”);
- **Hotelling-Prov., 18:19-29** (“Fig. 18 is an example of one embodiment of a main menu GUI 260 of a multi-functional device. As shown, the *GUI 260 includes icons/buttons 262 for launching each of the various device functionalities*. In this particular example, the main menu page 260 includes a PDA button 262A, *a cell phone button 262B, a music player button 262C*, a game player button 262D, a video player button 262E, a GPS button 262F, a remote control button 262G, a camera button 262H and a handtop button 262I. *The various buttons 262 are virtual buttons. When the button is pressed, the main page for the selected icon as shown in Figs. 9-17 is brought into view on the display. As an example, in order to select another device, the user simply selects a soft home button 264 located in the GUI of each device in order to load the main menu page 260*, and thereafter selects the desired device in the main menu page 260.”).
- **See also Hotelling-Prov., 16:32-17:5.**

185. See my discussion of elements [1.pre]–[1.a]. As explained for [1.pre]–[1.a] above, Hotelling discloses a first work process, which plays music. See [1.pre]–[1.a]. Hotelling further explicitly teaches that its “Music Player” is “based on the MP3 . . . format,” which discloses this claim. Hotelling-Prov., 8:29-9:2, 18:19-29.

(I) Claim [7]

186. **Claim [7]** recites “The method of claim 3, wherein the third command and the fifth command are inputted through at least one of a

predetermined key, a user's voice, and a motion of the mobile terminal.” This element is met by Hotelling.

187. **Hotelling discloses wherein the third command and the fifth command are inputted through at least one of a predetermined key, a user's voice, and a motion of the mobile terminal** (see [1.f], [3]; e.g., third and fifth commands input by “shaking” or using “voice”). See, e.g., Hotelling-Prov. (emphasis added):

- **Hotelling-Prov 19:9-16** (“*The command signal may also be generated by gestures initiated on the touch screen.* For example, sliding a finger (or stylus) across the display may cause a new page to be displayed. If slid to the right, the next page may be displayed. If slid to the left, the previous page may be displayed.
The command signal may also be generated by 3D device gestures created when the entire hand-held device is moved spatially. By way of example, *shaking the device may cause a new page to be displayed.*”);
- **Hotelling-Prov 19:21-30** (“The command signal may also be generated by sensing the orientation of the device. If the device is at 90 degrees, a first page is displayed, at 90 degrees a second page is displayed, at 180 degrees a third page is displayed and at 270 degrees a fourth 25 page is displayed.
The command signal may also be generated by monitoring a user's voice (i.e., voice recognition). If the user calls out ‘Phone’, the page associated with the phone is displayed, if the user calls out PDA, the page associated with the PDA is displayed.”).
- **See also Hotelling-Prov., 18:31-19:7, 19:18-25, 34:15-23.**

188. See my discussion of element [1.f] and claim [3]. As explained for [1.f] and [3] respectfully, Hotelling discloses a third command for activating a control menu and a fifth command for deactivating the control menu. See [1.f],

[3]. Hotelling further discloses generating the “command signal” through the “user’s voice” or by moving or “shaking” the device. Hotelling-Prov., 19:14-16, 19:21-25, 19:21-30. This not only renders obvious, but affirmatively discloses inputting commands using voice or motion of the device to activate control menus and deactivate control menus, as recited in element [1.f] and claim 3, respectively. For example, if a user calls out a “Phone” command or uses a shaking gesture or other similar gesture, the generated command signal will cause the associated “Phone” page to be displayed. See Hotelling-Prov., 19:21-30. And calling out a “close” command or shaking the phone or using some other gesture would have likewise closed and deactivated control menus. See Hotelling-Prov., 19:21-30.

E. Ground 6: Hotelling in view of Hellyar Renders Obvious Claims 1–4, 6–10

1. Overview of Hellyar and Motivation to Modify Hotelling with Hellyar’s Teachings

189. Hellyar is titled “Method and system for switching between multiple computer applications” and teaches allowing a user to view active applications as windows that display an “icon,” “text,” and a “preview representing a screen shot” of the application. Hellyar [0013], [0024], Fig. 3C. Hellyar’s user interface, presenting these windows, is provided on a “hand-held device[.]” Hellyar [0027]. By viewing active applications windows, including “icons” and a “preview,” the

user is able to “navigate through the icons” sequentially and “select an application” for “full viewing.” Hellyar [0003], [0024].

190. Hellyar teaches multiple methods of presenting the windows to the user for easily navigating through application windows. For example, Figures 3B and 3C show two embodiments of the presentation of the windows. In Figure 3B, the “windows are presented concurrently” and are “arranged in a circular fashion” with the active window “presented at the top position that would represent twelve o’clock on a clock face.” Hellyar [0044]. To navigate through the “remaining windows,” the user “presses the Tab key with the Alt key depressed,” which “cycle[s] through the windows” sequentially. Hellyar [0044]. Figure 3C shows a second embodiment:

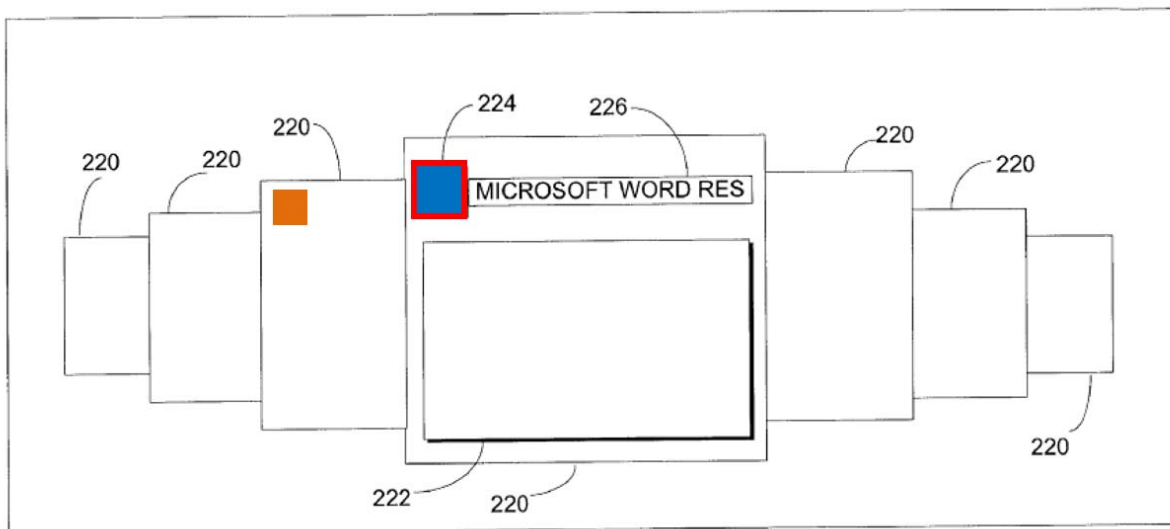


FIG. 3C.

Hellyar, Fig. 3C. Figure 3C shows a display where the user is shown windows that are “presented in a linear fashion,” with the “selected window” presented “in the center of the display on top.” Hellyar [0045]. As in the previous embodiment each window includes “a preview 222, an icon 224, and text 226.” Hellyar [0045]. An “**icon**” for the “selected window” is shown in **the top-left corner**, and since each preview window includes icons for the application represented, an “**icon**” is also displayed for the “next window” in the “activation hierarchy” immediately to the left of the center active window. Hellyar [0045]. In order to “cycle through the windows,” the user again “presses the Tab key with the Alt key depressed” to shift the windows from left to right—causing the “next window” to shift into the center of the display. Hellyar [0045].

191. Hotelling and Hellyar are analogous to and in the same field as the ’631 patent, which includes mobile interfaces (*e.g.*, user interfaces on mobile devices), and are reasonably pertinent to the alleged problems identified in the ’631 patent, *e.g.*, providing an interface for multi-tasking mobile terminals. See Section IX.B.1; ’631 patent, 2:32-51 (“In the conventional method for selecting the work process, when selecting a work process that is concurrently performed, the user has to select a new work process through a displayed menu window. Also, in order to terminate a predetermined work process among a plurality of work processes that are concurrently performed and select a new work process, the user has to select

the new work process through the menu window.”); Hotelling-Prov. 3:23-25 (“[A]n improved user interface that works for multi-functional devices.”); Hellyar [0003] (“[T]he present invention relates to computer software, and more particularly, to a method and system that allows users to navigate quickly through multiple applications.”), [0013] (“[A] method is provided for use in a computer system that allows users to navigate quickly through multiple applications.”).

192. Hotelling teaches “flipping (or scrolling) through the various GUIs” for different device functionalities. Hotelling-Prov, 18:31-19:3. Hellyar further discloses detailed teachings of including icons when flipping through GUIs to identify and preview each GUI for the user. *See* Hellyar [0013], [0024], [0044]-[0045]. Hellyar discloses the method “extracts a preview representing” the application and “displays the preview. . . and an icon for the next window in the activation hierarchy and . . . an icon for the remaining open” applications so “the user can navigate through the icons, viewing the preview for each icon.” Hellyar [0013], [0024]. A person of ordinary skill in the art would have been motivated to modify Hotelling with Hellyar’s teachings of including icons in the application previews when a user is cycling between them to advantageously improve the accessibility of the application switcher by providing clear labeling and identification for different windows in the application switcher. This would have

yielded a system that is sequentially displaying icons while flipping through GUIs for each application.

193. First, Hellyar discloses including icons as a part of the preview windows to provide additional information to the user, which in turn improves the user's ability to more conveniently and quickly "navigate quickly through multiple applications." Hellyar [0011]-[0013]. When a signal is received to display the windows for navigation, the system "extracts a preview representing a screen shot for each window open within the operating system." Hellyar [0013]. It then "displays the preview, a description, and an icon for the next window in the activation hierarchy and a description and an icon for the remaining open windows." Hellyar [0013]. The "icons are displayed" for "all open windows" for easy navigation. Hellyar [0013].

194. Second, Hotelling includes only broad disclosures of "flipping (or scrolling) through" GUIs, but leaves it to a person of ordinary skill in the art to implement the interface for flipping or scrolling. Hotelling-Prov. 18:31-19:3. Hellyar discloses in detail a specific embodiment of how to arrange, display, and incrementally cycle through previews, such as GUIs, which improves the accessibility of Hotelling's flipping and scrolling teachings. Hellyar [0024], [0044]-[0045]. Hellyar "present[s] in several different ways," the arrangement and display of the windows and icons for the user. Hellyar [0043]. For example, in one

embodiment, the “the previews, icons, and text are arranged in a circular fashion with the selected window presented at the top position of the circle or the position that would represent twelve o’clock on a clock face” and in a second embodiment, “[t]he previews, icons and text are presented in a linear fashion” with the “selected window . . . presented in the center of the display.” Hellyar [0044], [0045]. The different arrangements apply the same method of cycling through the previews and icons: “press[ing] the Tab button with the Alt key depressed to cycle through.” Hellyar [0045].

195. A person of ordinary skill in the art would have had a reasonable expectation of success when modifying Hotelling with Hellyar’s teachings because of the well-understood nature of these teachings. Indeed, it was well known in the art to scroll through sets of icons to navigate between applications or functions in a device. *E.g.*, Tu [0032] (“The effects processor 244 in the display control unit 204 manipulates the medium icon array 304 in the menu screen 302 by scrolling the medium icons in a horizontal direction. Since the medium icons 312-324 in the medium icon array 304 are organized in a circular database, every medium icon in the medium icon array 304 can be selected and displayed by the effects processor 244 by continuously scrolling in one direction.”); Lee (Ex. 1022), [0027] (“A menu display control method in an electronic device according to the present invention is a menu display control method in an electronic device comprising . . . a first

operation key for shifting a plurality of menu icons displayed on the display section, wherein the display section displays at least a section of the plurality of menu icons and a selection frame with a fixed position on the display section, and wherein the plurality of menu icons are arranged to enable virtual cyclical shifting.”).

2. Invalidity of Claims 1–4, 6–20 over Hotelling in view of Hellyar

(a) Element [1.a]

196. **Element [1.a] recites “displaying one icon in a designated portion of a display, wherein the displayed icon corresponds to the first work process.”** This element is met by Hotelling in view of Hellyar.

197. In addition to the teachings from Hotelling that I discussed above in Section IX.D.3(b), **Hellyar discloses displaying one icon in a designated portion of a display** (*e.g.*, “icon” within the corner of the “selected window ... in the center of the display”), **wherein the displayed icon corresponds to the first work process** (*e.g.*, “icon” represents an “open application[] window[]”). *See, e.g.*, Hellyar (emphasis added):

- **Hellyar, [0024]** (“The present invention provides a method and system for quickly and easily navigating between applications on a graphical operating system. ***Broadly, the invention monitors for user input indicative of a desire to navigate between open applications windows. If the input is received, a new window is displayed that contains an icon, text and a preview of the next window in the activation hierarchy.*** The icons for all other open windows are also displayed within the new window. ***The user***

can then navigate through the icons, viewing the preview for each icon selected. Upon receiving input indicative of a desire to select an application window for full viewing, the invention brings the selected icon's corresponding window to the foreground and the new window is no longer displayed. Having briefly described an embodiment of the present invention, an exemplary operating environment for the present invention is described below.”);

- **Hellyar, [0040]** (“Returning to FIG. 2, if it is determined at step 206 that the navigation preview is not already displayed, then at step 208 the top-level windows are enumerated. Next, at step 210, a preview, an icon and window text for each open window are extracted. The invention thus has the information needed for the navigation display after step 210. The next window in the activation hierarchy is then selected at step 212. If it is determined at step 206 that the previews are already showing, steps 208 and 210 are skipped and the next window in the activation hierarchy is then selected at step 212. In another embodiment, the user may input the Alt+Shift+Tab keystroke so that the previous window in the activation hierarchy is activated. *In a further embodiment, instead of the Alt+Tab keystroke, a user may indicate the icon he or she wants to select by using a pointing device, such as a mouse.* At step 214, the preview, icon and text for at least the window selected in step 212 is displayed along with icons from each of the other open top-level windows. *As more fully described below, depending on the navigation display, the preview, icon and text for each open window may be arranged and displayed with the window selected in step 212 highlighted in some fashion.* The system then returns to monitoring the keyboard input at step 202.”);
- **Hellyar, [0042]** (“ In use, the invention provides more useful information to a user allowing the user to quickly and efficiently navigate to a desired open window. The user can activate the invention by initiating the navigation key sequence, such as by depressing the Alt and Tab keys. Upon receiving the navigation key sequence in the first instance, *the invention will display at least the icons associated with each open window as well as text and a preview of the next window in the activation hierarchy.* As long as the user keeps the Alt key depressed the information remains displayed. *Successively pressing the Tab key with the Alt key depressed iterates through the activation hierarchy such that successive previews are highlighted.* When the user has navigated to the desired window, the Alt key is released to bring the selected window to the forefront.”);

- **Hellyar, [0045]** (*“In FIG. 3C, the windows 220 having previews 222, icons 224 and text 226 are again presented concurrently. Although not shown in FIG. 3C, it should be understood that each window 220 contains a preview 222, an icon 224 and text 226. The previews, icons and text are presented in a linear fashion. The selected window is presented in the center of the display on top of the other windows and is larger than the other windows. A portion of the remaining windows as enumerated in step 208 are presented to the left and right of the selected window with the next window in the activation hierarchy immediately to the left of the selected window. The size of the window decreases as the position of the window moves away from the selected center window. When the user presses the Tab button with the Alt key depressed to cycle through the windows, the windows shift so that the selected window is always in the center of the display.”*).
- **See also Hellyar, Fig. 3B, [0041], [0044].**

198. In addition to Hotelling’s teachings, as discussed in Section IX.D.3(b), Hellyar discloses “navigating between applications” on a device by cycling through navigation application windows with “previews, icons, and text.” Hellyar [0024], [0045]. As shown in Figure 3 below, “windows 220” include “previews 222, icons 224 and text 226”:

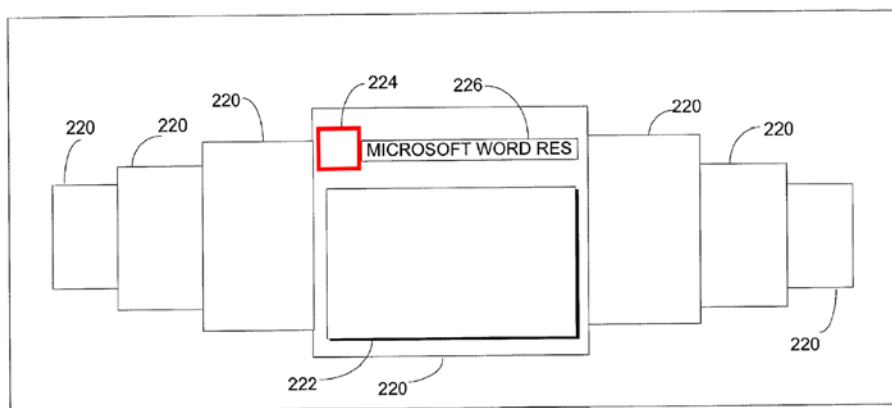


FIG. 3C.

Hellyar, Fig. 3, [0045]. Icons 224 are displayed in a designated portion of the display at the top-left corner of the central window. Hellyer, Fig. 3, [0045]. As discussed in Section IX.D.3(b), a person of ordinary skill in the art would have been motivated to modify Hotelling’s teachings of flipping or scrolling through GUIs with Hellyar’s teachings of presenting associated “previews, icons, and text” in the designated portion of each application window. This would have yielded an interface in which each of the application windows has an associated icon 224 as taught by Hellyar.

(b) Element [1.b]

199. **Element [1.b] recites “if the displayed icon corresponding to the first work process is selected, displaying a screen of the first work process in the display.”** This element is met by Hotelling in view of Hellyar.

200. In addition to the teachings from Hotelling that I discussed above in Section IX.D.3(c), **Hellyar discloses if the displayed icon corresponding to the first work process is selected** (*see* [1.a]; “receiving input indicative of a desire to select an application”), **displaying a screen of the first work process in the display** (*e.g.*, “the top-level window corresponding to the selected preview and icon are brought to the foreground”). *See, e.g.*, Hellyar (emphasis added):

- **Hellyar, [0024]** (“The present invention provides a method and system for quickly and easily navigating between applications on a graphical operating system. ***Broadly, the invention monitors for user input indicative of a***

desire to navigate between open applications windows. If the input is received, a new window is displayed that contains an icon, text and a preview of the next window in the activation hierarchy. The icons for all other open windows are also displayed within the new window. *The user can then navigate through the icons, viewing the preview for each icon selected. Upon receiving input indicative of a desire to select an application window for full viewing, the invention brings the selected icon's corresponding window to the foreground and the new window is no longer displayed.* Having briefly described an embodiment of the present invention, an exemplary operating environment for the present invention is described below.”);

- **Hellyar, [0040]** (“Returning to FIG. 2, if it is determined at step 206 that the navigation preview is not already displayed, then at step 208 the top-level windows are enumerated. Next, at step 210, a preview, an icon and window text for each open window are extracted. The invention thus has the information needed for the navigation display after step 210. The next window in the activation hierarchy is then selected at step 212. If it is determined at step 206 that the previews are already showing, steps 208 and 210 are skipped and the next window in the activation hierarchy is then selected at step 212. In another embodiment, the user may input the Alt+Shift+Tab keystroke so that the previous window in the activation hierarchy is activated. *In a further embodiment, instead of the Alt+Tab keystroke, a user may indicate the icon he or she wants to select by using a pointing device, such as a mouse.* At step 214, the preview, icon and text for at least the window selected in step 212 is displayed along with icons from each of the other open top-level windows. *As more fully described below, depending on the navigation display, the preview, icon and text for each open window may be arranged and displayed with the window selected in step 212 highlighted in some fashion.* The system then returns to monitoring the keyboard input at step 202.”);
- **[0041]** (“Returning to step 202, if the Alt+Tab key sequence is not received, it is determined at step 216 whether the navigation preview is already showing and the Alt key has been released. If the previews are not showing, the system returns to step 202 to monitor the keyboard input. This addresses the scenario where the user has not yet indicated a desire to navigate among the open windows. Further, if the previews are showing and the Alt key has not been released, the system also returns to step 202 to monitor the keyboard input. This addresses the scenario where the user is still in a

navigational mode but has not indicated a desire to select a different window. ***If the previews are showing and the Alt key has been released, the system proceeds to step 218. At step 218, the top-level window corresponding to the selected preview and icon are brought to the foreground and the previews are dismissed.***”).

- ***See also Hellyar [0008], [0032], [0042].***

201. In addition to Hotelling’s teachings, as discussed in Section IX.D.3(c), Hellyar also discloses that each application window for a work process is displayed with an icon in a designated location above the preview for that application. See [1.a]. Further, Hellyar discloses a user “select[ing] an application,” including by selecting its corresponding icon, to “full[y] view[]” the application in the display foreground. Hellyar [0024], [0041]. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling with Hellyar’s teachings of application windows with icons to advantageously improve the ease of operation of this application switcher feature, rendering obvious this element.

(c) Element [1.c]

202. **Element [1.c] recites “in response to a first command for sequentially displaying icons, performing sequential display of icons such that an icon corresponding to the second work process is displayed, wherein the icon corresponding to the second work process is displayed in the designated portion of the display.”** This element is met by Hotelling in view of Hellyar.

203. In addition to the teachings from Hotelling that I discussed above in Section IX.D.3(d), **Hellyar discloses in response to a first command for sequentially displaying icons, performing sequential display of icons such that an icon corresponding to the second work process is displayed (e.g., “[s]uccessively pressing the Tab key” to “iterate[] through” program “previews” such that “windows shift” in the display), wherein the icon corresponding to the second work process is displayed in the designated portion of the display (see [1.a]; icon is displayed the top-left corner of the “selected window” in the center).** See, e.g., Hellyar (emphasis added):

- **Hellyar, Fig. 3C**

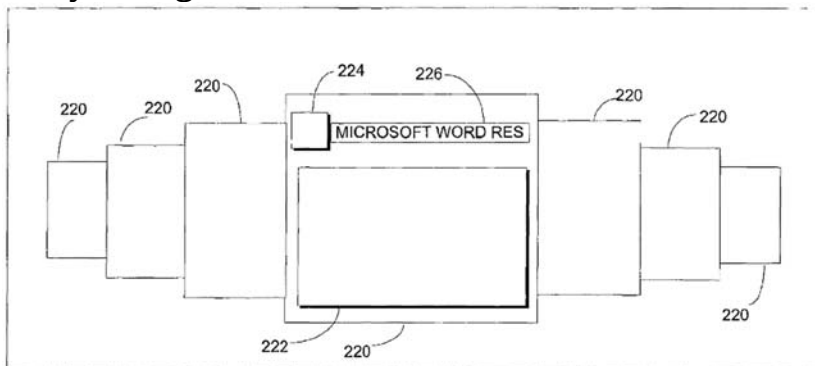


FIG. 3C.

- **Hellyar, [0024]** (“The present invention provides a method and system for quickly and easily navigating between applications on a graphical operating system. ***Broadly, the invention monitors for user input indicative of a desire to navigate between open applications windows. If the input is received, a new window is displayed that contains an icon, text and a preview of the next window in the activation hierarchy.*** The icons for all other open windows are also displayed within the new window. ***The user can then navigate through the icons, viewing the preview for each icon selected. Upon receiving input indicative of a desire to select an***

application window for full viewing, the invention brings the selected icon's corresponding window to the foreground and the new window is no longer displayed. Having briefly described an embodiment of the present invention, an exemplary operating environment for the present invention is described below.”);

- **Hellyar, [0042]** (“ In use, the invention provides more useful information to a user allowing the user to quickly and efficiently navigate to a desired open window. The user can activate the invention by initiating the navigation key sequence, such as by depressing the Alt and Tab keys. Upon receiving the navigation key sequence in the first instance, *the invention will display at least the icons associated with each open window as well as text and a preview of the next window in the activation hierarchy.* As long as the user keeps the Alt key depressed the information remains displayed. *Successively pressing the Tab key with the Alt key depressed iterates through the activation hierarchy such that successive previews are highlighted.* When the user has navigated to the desired window, the Alt key is released to bring the selected window to the forefront.”);
- **Hellyar, [0045]** (“*In FIG. 3C, the windows 220 having previews 222, icons 224 and text 226 are again presented concurrently. Although not shown in FIG. 3C, it should be understood that each window 220 contains a preview 222, an icon 224 and text 226. The previews, icons and text are presented in a linear fashion. The selected window is presented in the center of the display on top of the other windows and is larger than the other windows. A portion of the remaining windows as enumerated in step 208 are presented to the left and right of the selected window with the next window in the activation hierarchy immediately to the left of the selected window.* The size of the window decreases as the position of the window moves away from the selected center window. *When the user presses the Tab button with the Alt key depressed to cycle through the windows, the windows shift so that the selected window is always in the center of the display.*”).
- **See also Hellyar, Fig. 3B, [0032], [0038], [0040], [0044].**

204. In addition to Hotelling’s teachings, as discussed in Section IX.D.3(d), Hellyar discloses a command for sequentially displaying icons and performing sequential display of icons such that an icon corresponding to a work

process is displayed in the designated portion. Hellyar discloses that pressing a “Tab” key will sequentially and “successive[ly]” “cycle” through the application windows shown in Figure 3. Hellyar, Fig. 3, [0042], [0045]. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling’s application switcher within its GUI with Hellyar’s cycling application windows teachings in order to advantageously improve the ease of accessibility of the application switcher by providing clear labels and identifications for the different windows in the application switcher. Such a system would yield a GUI in which, for example, Hotelling’s “cell phone” icon is “incrementally” brought into view together with the “cell phone” application window in the application switcher in response to user input to cycle through application windows. See Hotelling-Prov., 18:19-19:3; Hellyar [0042], [0045], Fig. 3C.

(d) Element [1.d]

205. **Element [1.d] recites “in response to a second command for changing to the second work process, replacing the screen of the first work process with a screen of the second work process.”** This element is met by Hotelling in view of Hellyar.

206. In addition to the teachings from Hotelling that I discussed above in Section IX.D.3(e), **Hellyar discloses in response to a second command for**

changing to the second work process (e.g., “input indicative of a desire to select an application window for full viewing,”), **replacing the screen of the first work process with a screen of the second work process** (e.g., “bring[ing] the selected icon’s corresponding window to the foreground,” replacing the previous screen).

See, e.g., Hellyar (emphasis added):

- **Hellyar, [0024]** (“The present invention provides a method and system for quickly and easily navigating between applications on a graphical operating system. *Broadly, the invention monitors for user input indicative of a desire to navigate between open applications windows. If the input is received, a new window is displayed that contains an icon, text and a preview of the next window in the activation hierarchy.* The icons for all other open windows are also displayed within the new window. *The user can then navigate through the icons, viewing the preview for each icon selected. Upon receiving input indicative of a desire to select an application window for full viewing, the invention brings the selected icon's corresponding window to the foreground and the new window is no longer displayed.* Having briefly described an embodiment of the present invention, an exemplary operating environment for the present invention is described below.”);
- **Hellyar, [0040]** (“ Returning to FIG. 2, if it is determined at step 206 that the navigation preview is not already displayed, then at step 208 the top-level windows are enumerated. Next, at step 210, a preview, an icon and window text for each open window are extracted. The invention thus has the information needed for the navigation display after step 210. The next window in the activation hierarchy is then selected at step 212. If it is determined at step 206 that the previews are already showing, steps 208 and 210 are skipped and the next window in the activation hierarchy is then selected at step 212. In another embodiment, the user may input the Alt+Shift+Tab keystroke so that the previous window in the activation hierarchy is activated. *In a further embodiment, instead of the Alt+Tab keystroke, a user may indicate the icon he or she wants to select by using a pointing device, such as a mouse.* At step 214, the preview, icon and text for at least the window selected in step 212 is displayed along with icons

from each of the other open top-level windows. ***As more fully described below, depending on the navigation display, the preview, icon and text for each open window may be arranged and displayed with the window selected in step 212 highlighted in some fashion.*** The system then returns to monitoring the keyboard input at step 202.”);

- **Hellyar, [0042]** (“ In use, the invention provides more useful information to a user allowing the user to quickly and efficiently navigate to a desired open window. The user can activate the invention by initiating the navigation key sequence, such as by depressing the Alt and Tab keys. Upon receiving the navigation key sequence in the first instance, ***the invention will display at least the icons associated with each open window as well as text and a preview of the next window in the activation hierarchy.*** As long as the user keeps the Alt key depressed the information remains displayed. ***Successively pressing the Tab key with the Alt key depressed iterates through the activation hierarchy such that successive previews are highlighted.*** When the user has navigated to the desired window, the Alt key is released to bring the selected window to the forefront.”).
- **See also Hellyar [0038], [0044]-[0045].**

207. In addition to Hotelling’s teachings, as discussed in Section IX.D.3(e), Hellyar discloses work processes and in response commands for changing to a work process, replacing the screen of one work process with the screen of another work process. See [1.a]–[1.c]. Hellyar further teaches that in response to user input “select[ing] an application window for full viewing,” the device opens that application. Hellyar [0024], [0042], [0045]. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling’s GUI switcher teachings with Hellyar’s teachings of including icons with application windows. For example, in such system, each time a “tab” key is pressed—or a swipe or other command is issued to provide a command to cycle

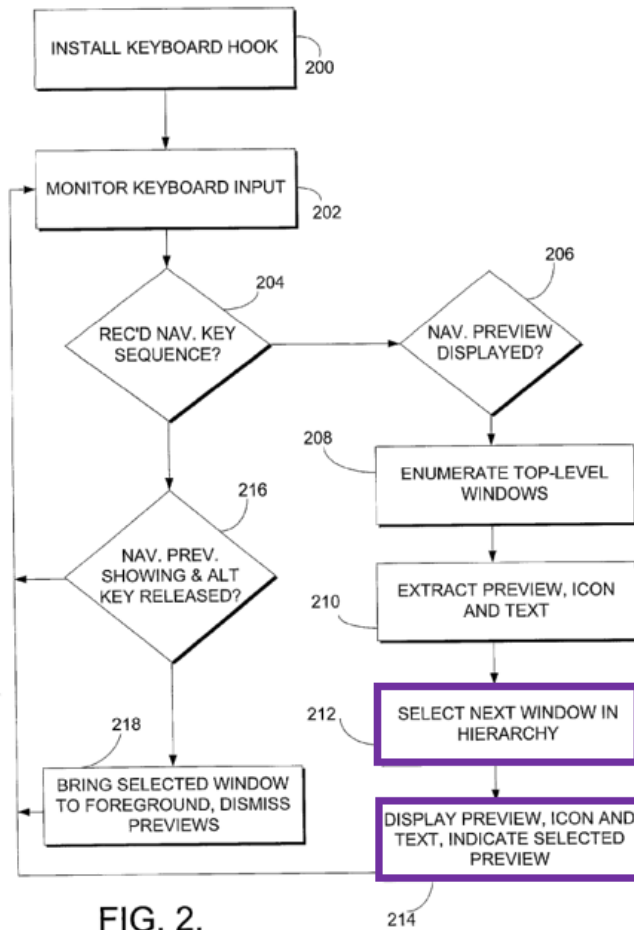
between GUIs—this results in the applications, and the associated icons, to swipe through the display. When a user then selects an icon, such as Hotelling’s “cell phone” icon, in an application window, as taught by Hellyar, this would cause the cell phone application to replace the screen of whichever application is currently-open. Hotelling-Prov., 18:20-19:3.

(e) Claim [6]

208. **Claim [6] recites “The method of claim 1, wherein an item of the second work process is displayed relatively brighter or clearer than an item or a screen of a work process other than the second work process.”** This Claim is met by Hotelling in view of Hellyar.

209. **Hellyar discloses an item of the second work process is displayed relatively brighter or clearer than an item or screen of a work process other than the second work process** (*e.g.*, “the preview, icon and text” for the “selected” window is “highlighted”). *See, e.g.*, Hellyar (emphasis added):

- **Hellyar, Fig. 2**



- **Hellyar, Fig. 3C**

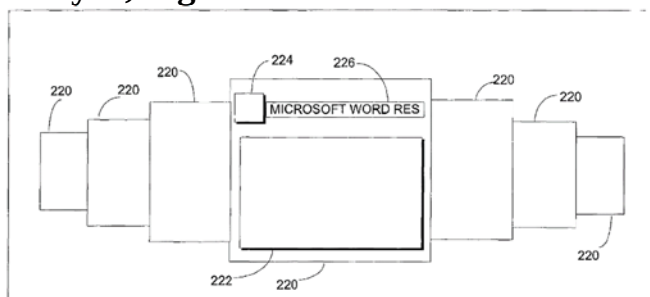


FIG. 3C.

- **Hellyar, [0040]** (“ Returning to FIG. 2, if it is determined at step 206 that the navigation preview is not already displayed, then at step 208 the top-level windows are enumerated. Next, at step 210, a preview, an icon and window text for each open window are extracted. The invention thus has the information needed for the navigation display after step 210. The next

window in the activation hierarchy is then selected at step 212. If it is determined at step 206 that the previews are already showing, steps 208 and 210 are skipped and the next window in the activation hierarchy is then selected at step 212. In another embodiment, the user may input the Alt+Shift+Tab keystroke so that the previous window in the activation hierarchy is activated. ***In a further embodiment, instead of the Alt+Tab keystroke, a user may indicate the icon he or she wants to select by using a pointing device, such as a mouse.*** At step 214, the preview, icon and text for at least the window selected in step 212 is displayed along with icons from each of the other open top-level windows. ***As more fully described below, depending on the navigation display, the preview, icon and text for each open window may be arranged and displayed with the window selected in step 212 highlighted in some fashion.*** The system then returns to monitoring the keyboard input at step 202.”).

- ***See also Hellyar, [0010], [0042], [0045].***

210. See my discussion of elements [1.pre]–[1.c]. As explained for [1.pre]–[1.c], Hellyar discloses an item of one work process—e.g., window 220, a preview 222, an icon 224, and text 226—and displaying a screen of another process. See [1.pre]–[1.c]. Hellyar further discloses while navigating between window, a “selected” window is “highlighted in some fashion.” Hellyar [0040], Fig. 2. For example, when an application’s window is the central window (e.g., a “Cell Phone” application as taught by Hotelling), a person of ordinary skill in the art would have found it obvious to “highlight[]” that window by displaying it brighter or clearer than the other application windows. Hellyar [0040]. Highlighting certain objects on screen such that they are brighter than other objects was known in the art as a technique used to make it clearer to a user to understand what was currently being selected or interacted with, and a person of ordinary skill

in the art would have found it obvious to make a selected or interacted with item brighter or clearer in order to make it easier for users to identify candidate objects for selection. *E.g.*, Marmolin (Ex. 1023), [0011]-[0012] (describing a “trackable spotlight” to make user interface “object[s] within said spotlight region brighter than its surroundings.”); Tu [0040] (describing changing “display parameters” such as “color, size, brightness, saturation and/or hue” for items moved into interface’s selection area).

211. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling’s GUI teachings with Hellyar’s application window teachings to advantageously draw attention to the selected GUI. For example, in such a system, information about an application window of interest would be displayed brighter and more clearly than the background, so that accessing context about the applications is more visually discernible for a user. *See* Hotelling-Prov., 18:31-19:7; Hellyar [0040].

(f) Claim [8]

212. **Claim [8] recites “The method of claim 1, wherein, in response to a repetition of the first command, performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal.”** This Claim is met by Hotelling in view of Hellyar.

213. Hellyar further discloses in response to a repetition of the first command (see [1.c]; e.g., repeating the command to “iterate[] through” program “previews”), performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal (e.g., see [1.c]; e.g., repeating the first command covers an “icon” as the next program is brought into view). See, e.g., Hellyar (emphasis added):

- Hellyar, Fig. 3C

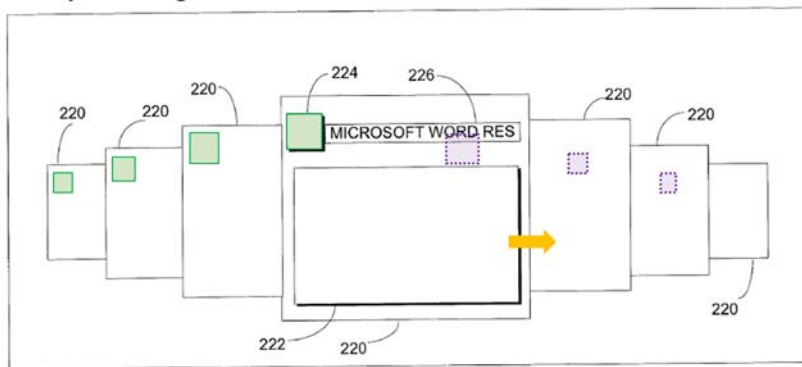


FIG. 3C.

- Hellyar, [0042] (“In use, the invention provides more useful information to a user allowing the user to quickly and efficiently navigate to a desired open window. The user can activate the invention by initiating the navigation key sequence, such as by depressing the Alt and Tab keys. Upon receiving the navigation key sequence in the first instance, ***the invention will display at least the icons associated with each open window as well as text and a preview of the next window in the activation hierarchy.*** As long as the user keeps the Alt key depressed the information remains displayed. ***Successively pressing the Tab key with the Alt key depressed iterates through the activation hierarchy such that successive previews are highlighted.*** When the user has navigated to the desired window, the Alt key is released to bring the selected window to the forefront.”);

- **Hellyar, [0045]** (*“In FIG. 3C, the windows 220 having previews 222, icons 224 and text 226 are again presented concurrently. Although not shown in FIG. 3C, it should be understood that each window 220 contains a preview 222, an icon 224 and text 226. The previews, icons and text are presented in a linear fashion. The selected window is presented in the center of the display on top of the other windows and is larger than the other windows. A portion of the remaining windows as enumerated in step 208 are presented to the left and right of the selected window with the next window in the activation hierarchy immediately to the left of the selected window. The size of the window decreases as the position of the window moves away from the selected center window. When the user presses the Tab button with the Alt key depressed to cycle through the windows, the windows shift so that the selected window is always in the center of the display.”*).
- **See also Hellyar Fig. 3B, [0044].**

214. See my discussion of elements [1.a] and [1.c]. As explained for [1.a] above, Hellyar discloses displaying icons corresponding to work processes. See [1.a]. As further explained for [1.c], Hellyar discloses sequentially displaying icons. See [1.c]. Hellyar further discloses that “[s]uccessively pressing the Tab key” or “Alt key” to display the next application will “iterate[]” through the application windows containing “icon 224.” Hellyar [0042], [0045]. As shown in Figure 3C, icons are obscured by the overlapping windows on top of them. See Hellyar, Fig. 3C. For example, the icons to the left of the central application window in Figure 3C (as well as the icon for the central application window itself) are visible, but the icons on windows to the right of the central window are not visible. See Hellyar, Fig. 3C. Thus, as the user cycles through the windows, to the right, icons corresponding to application windows cycling to the right of the central

window disappear from the screen while icons corresponding to application windows to the left of the central window appear on the screen. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling's device GUI with Hellyar's detailed cycling application window teachings in order to make application switching more accessible by providing more information about each application in an application window.

(g) Claim [9]

215. **Claim [9] recites “The method of claim 1, wherein an icon to be displayed in the designated portion is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command.”** This Claim is met by Hotelling in view of Hellyar.

216. **Hellyar discloses wherein an icon to be displayed in the designated portion is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command** (*e.g.*, see [1.a], [1.c]; “icons” “shift” right in response to a user command to “iterate[]” through the windows, such that the icon that will be displayed in the center is replaced with another icon). *See, e.g.*, Hellyar (emphasis added):

- Hellyar, Fig. 3C

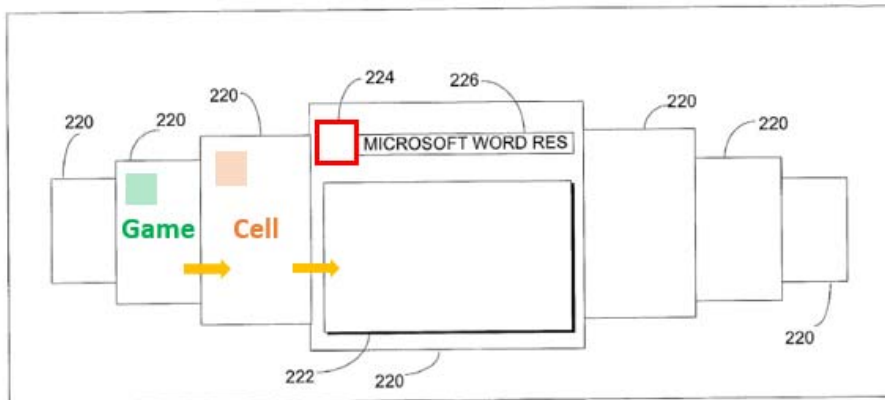


FIG. 3C.

- Hellyar, [0042] (“In use, the invention provides more useful information to a user allowing the user to quickly and efficiently navigate to a desired open window. The user can activate the invention by initiating the navigation key sequence, such as by depressing the Alt and Tab keys. Upon receiving the navigation key sequence in the first instance, ***the invention will display at least the icons associated with each open window as well as text and a preview of the next window in the activation hierarchy.*** As long as the user keeps the Alt key depressed the information remains displayed. ***Successively pressing the Tab key with the Alt key depressed iterates through the activation hierarchy such that successive previews are highlighted.*** When the user has navigated to the desired window, the Alt key is released to bring the selected window to the forefront.”);
- Hellyar, [0045] (“In FIG. 3C, the windows 220 having previews 222, icons 224 and text 226 are again presented concurrently. Although not shown in FIG. 3C, it should be understood that each window 220 contains a preview 222, an icon 224 and text 226. The previews, icons and text are presented in a linear fashion. The selected window is presented in the center of the display on top of the other windows and is larger than the other windows. A portion of the remaining windows as enumerated in step 208 are presented to the left and right of the selected window with the next window in the activation hierarchy immediately to the left of the selected window. The size of the window decreases as the position of the window moves away from the selected center window. ***When the user presses the Tab button with the Alt key depressed to cycle through the windows, the windows shift so that the selected window is always in the center of the display.***”).

- **See also Hellyar Fig. 3B, [0044].**

217. See my discussion of elements [1.a] and [1.c]. As explained for [1.a] above, Hellyar discloses an icon to be displayed in the designated portion. See [1.a]. As I further explained for [1.c] above, Hellyar discloses replacing icons to be displayed in the designated portion with an icon corresponding to a different work process. See [1.c]. As discussed for [1.a], Hellyar teaches displaying application icons in the top-left corner of each “application window.” Hellyar [0045], Fig. 3C. When “iterat[ing]” through the windows such that they shift right, as discussed for [1.c], the icon for the “next window in the activation hierarchy immediately to the left of the selected window” will shift over and become the icon shown in “selected” window in the center of the display. See Hellyar [0045]. In other words, **the icon for the next window in the hierarchy** (e.g., the icon for Hotelling’s “Cell Phone” shown in annotated Figure 3C) is an icon to be displayed in the **designated portion** of the screen.

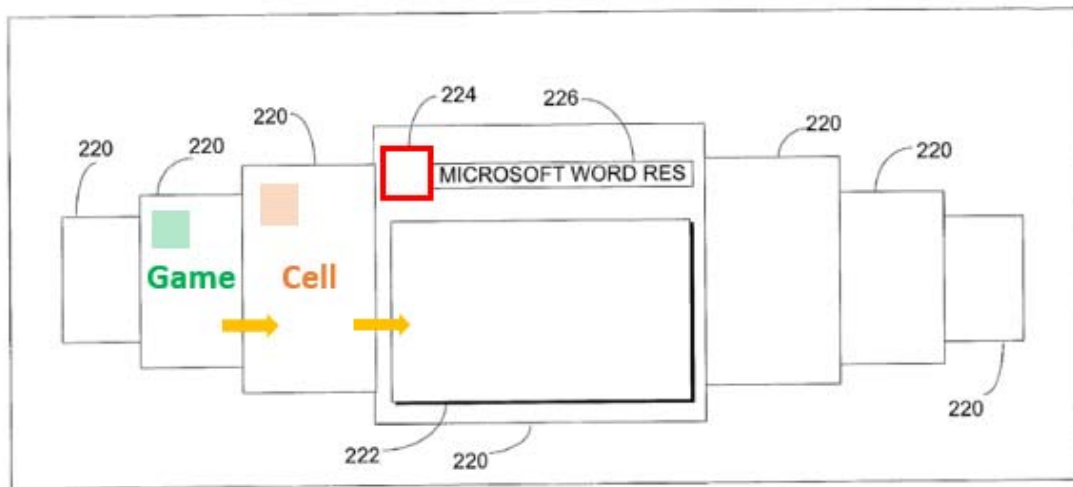


FIG. 3C.

At the same time, **an icon for a different application** (e.g., for a game, or some other application, as shown in annotated Figure 3C) will shift over and replace the **icon for the next window in the hierarchy**. Hellyar [0045]. In general, that new icon may correspond to any program available on the device, such as “a word processor, spreadsheet and/or internet browser,” “messaging and organization program[s],” for example. Hellyar [0004]. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling’s device GUIs with Hellyar’s teachings of navigating between applications using previews and icons.

(h) Claim [10]

218. **Claim [10] recites “The method of claim 1, wherein the displayed icon is displayed together with at least a part of a screen image of a work process in the display.”** This element is met by Hotelling in view of Hellyar.

219. Hellyar discloses the displayed icon is displayed together with at least a part of a screen image of a work process in the display (e.g., Figure 3C, “the windows 220” have “previews” and “icons” presented concurrently”). See, e.g., Hellyar (emphasis added):

- **Hellyar, Fig. 3C**

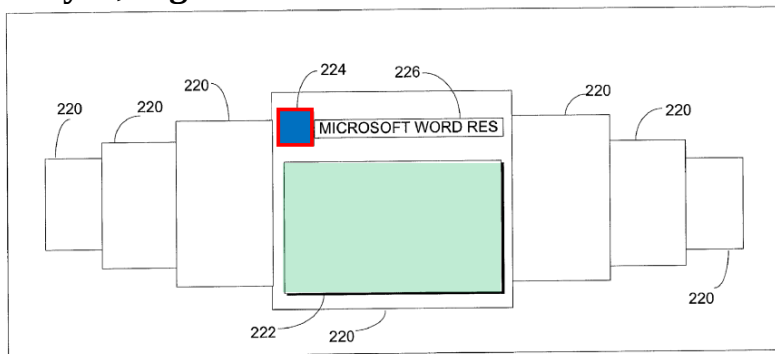


FIG. 3C.

- **Hellyar, [0013]** (“ Generally described, a method is provided for use in a computer system that allows users to navigate quickly through multiple applications. In accordance with the method, when the computer receives a signal from the user, *it extracts a preview representing a screen shot for each window open within the operating system. It then displays the preview, a description, and an icon for the next window in the activation hierarchy* and a description and an icon for the remaining open windows. In other embodiments, the previews for all open windows with corresponding descriptions and icons are displayed utilizing several different arrangements. The computer then receives a designation from the user of the next window he or she wants to make active and brings to the foreground the top level window corresponding to the selection.”);
- **Hellyar, [0045]** (“*In FIG. 3C, the windows 220 having previews 222, icons 224 and text 226 are again presented concurrently. Although not shown in FIG. 3C, it should be understood that each window 220 contains a preview 222, an icon 224 and text 226. The previews, icons and text are presented in a linear fashion. The selected window is presented in the center of the display on top of the other windows and is larger than the other windows. A portion of the remaining windows as enumerated in step 208 are*

presented to the left and right of the selected window with the next window in the activation hierarchy immediately to the left of the selected window.

The size of the window decreases as the position of the window moves away from the selected center window. ***When the user presses the Tab button with the Alt key depressed to cycle through the windows, the windows shift so that the selected window is always in the center of the display.”).***

- **See also Hellyar [0050].**

220. See my discussion of elements [1.a] and [1.c]. As explained for [1.a] above, Hellyar discloses displaying icons corresponding to work processes. See [1.a]. As I further explained for element [1.c] above, Hellyar discloses displaying a screen image of a work process. See [1.c]. Hellyar further discloses that the application window includes the application “preview, icon and text.” Hellyar [0040]. The preview is a “screen shot” of the open application. Hellyar [0013], [0045]. Thus, the application icon is displayed together with a part, or screen shot” of the application. As discussed in Section IX.E.1, a person of ordinary skill in the art would have been motivated to modify Hotelling’s device GUIs with Hellyar’s teachings of navigating between applications using previews and icons .

F. Ground 7: Hotelling in view of Hellyar and Tu Renders Obvious Claims 5–6

1. Overview of Tu and Motivation to Modify Hotelling in view of Hellyar with Tu’s Teachings

221. Tu is titled “Control of background media when foreground graphical user interface is invoked” and teaches that a “menu” is overlaid on a “dim[med]” background. Tu [0045]. Tu teaches that a “foreground GUI menu screen” is

“superimposed on top” of media content, such as music playing in a music player, that is active in the background and the media content in the background is “dim[med]” to “contrast the foreground GUI menu screen from the background.”

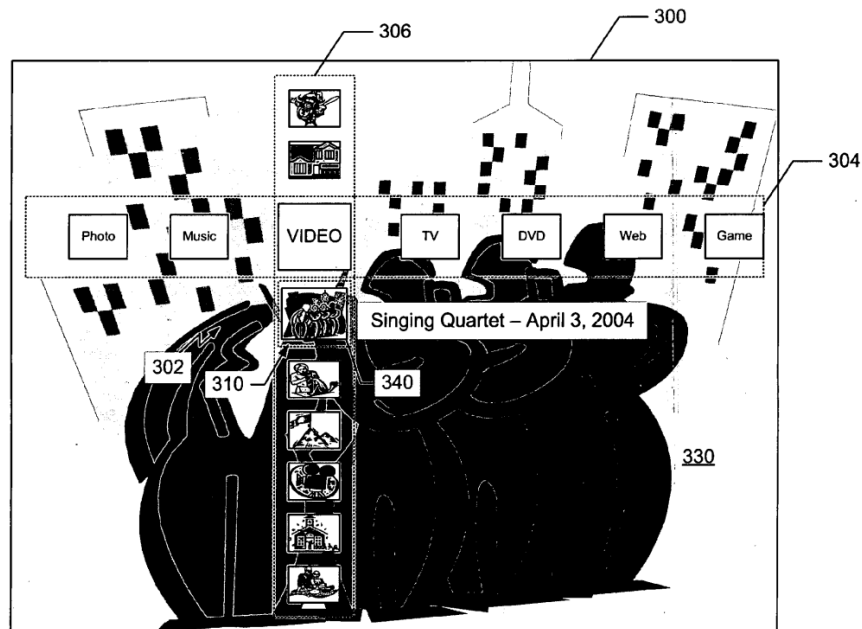


FIG. 4

Tu, Figs. 3-4, [0033], [0042], [0045], [0049]-[0050].

222. Tu teaches that in order to distinguish certain items for “easy viewing,” these items in a foreground menu are positioned in the “**center area 308**” or the “attention area 310” of a navigation menu and are displayed brighter and sometimes “flashing.”

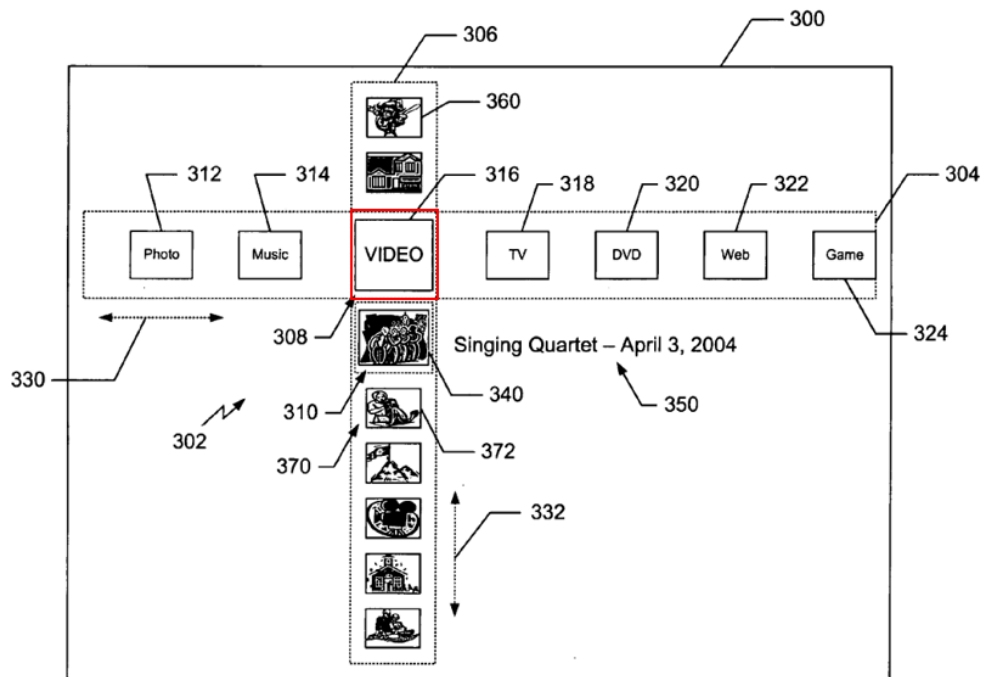


FIG. 3

Tu, Fig. 3; [0033], [0039]-[0040].

223. Hotelling, Hellyar, and Tu are analogous to and in the same field as the '631 patent, which includes mobile interfaces (*e.g.*, user interfaces on mobile devices), and are reasonably pertinent to the alleged problems identified in the '631 patent, which includes providing a display for a multi-tasking mobile terminal. See Section IX.B.1; '631 patent, 2:32-51 ("In the conventional method for selecting the work process, when selecting a work process that is concurrently performed, the user has to select a new work process through a displayed menu window. Also, in order to terminate a predetermined work process among a plurality of work processes that are concurrently performed and select a new work

process, the user has to select the new work process through the menu window.”); Hotelling-Prov. 3:23-25 (“[A]n improved user interface that works for multi-functional devices.”); Hellyar [0003] (“[T]he present invention relates to computer software, and more particularly, to a method and system that allows users to navigate quickly through multiple applications.”), [0013] (“[A] method is provided for use in a computer system that allows users to navigate quickly through multiple applications.”); Tu [0005] (“The present invention includes . . . control of multimedia contents displayed in the background when the foreground graphical user interface (GUI) is invoked.”).

224. A person of ordinary skill in the art would have been motivated to modify Hotelling’s (as modified by Hellyar) GUI switcher teachings with Tu’s teachings of displaying a menu of icons with a bright central icon in the foreground overlaid on a dimmed background. Tu discloses that traditional interfaces with menus and icons are difficult to use when media is playing in the background. Tu [0004]-[0008]. Tu teaches that modifying the background and the icons in the foreground menu makes it “easy” to view the menu and the selected preview window, while still allowing the user to see the media content being displayed in the background. Tu Figs 3-4, [0033], [0039]-[0040]; Hellyar, Fig. 3, [0045]. See Tu [0033].

225. A person of ordinary skill in the art would have had a reasonable expectation of success modifying Hotelling in view of Hellyar with Tu’s teachings, yielding a system in which the GUI switcher menu with icons, as taught by Hotelling in view of Hellyar, would be overlaid over a dimmed background, with the icon for the selected application being displayed brighter and/or flashing. Overlaying images and using brightness or clarity to advantageously visually distinguish particular screen elements was well-known in the art, and modifying Hotelling and Hellyar with Tu’s teachings is merely applying this known concept for its intended purpose. Marmolin, [0011]-[0012] (describing a “trackable spotlight” to make user interface “object[s] within said spotlight region brighter than its surroundings”); Gardien (Ex. 1024), 6:14-22 (disclosing “highlighting” an “option for selection” on a display screen menu); Runov (Ex. 1025), [0017] (disclosing “highlighting icons and items” on a computer display that match search criteria); Park (Ex. 1026), [0084] (disclosing showing certain icons “brighter in its background color . . . so as to be discriminated from other icons”).

2. Invalidity of Claims 5–6 Over Hotelling in view of Hellyar and Tu

(a) Claim [5]

226. **Claim [5] recites “The method of claim 1, wherein an item of the first work process is displayed relatively brighter or clearer than the screen of**

the first work process.” This Claim is met by Hotelling in view of Hellyar and Tu.

227. In addition to the teachings from Hotelling and Hellyar that I discussed above in Sections IX.D-IX.E, **Tu discloses an item of the first work process is displayed relatively brighter or clearer than the screen of the first work process** (e.g., music in background is “dim[med]” while “music” menu item is showing). See, e.g. Tu (emphasis added):

- **Tu, Figs. 3-4**

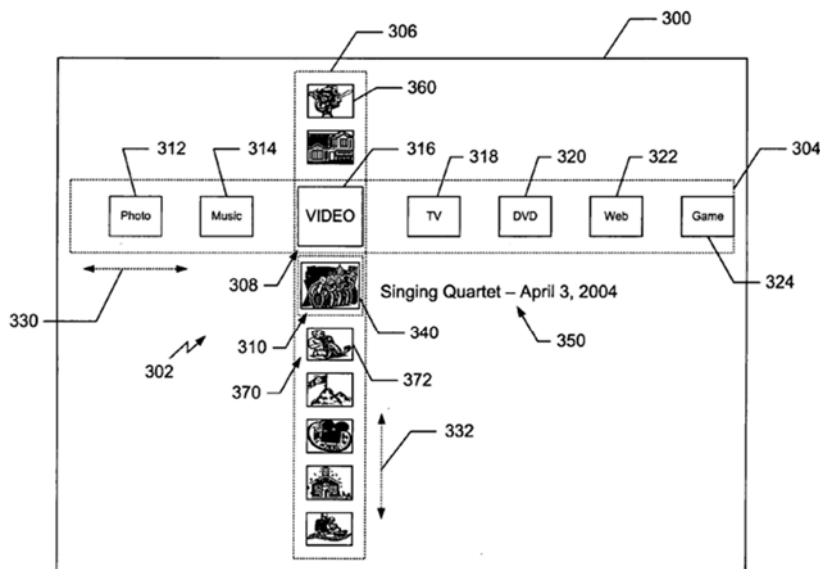


FIG. 3

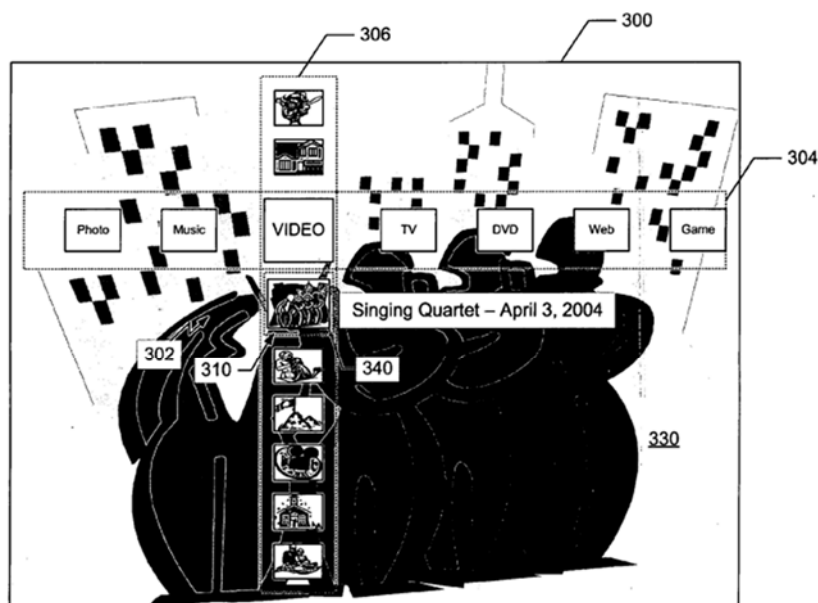


FIG. 4

- **Tu, [0033]** (“As described above, the effects processor 244 displays the medium icons with the same display parameters while the icons are being scrolled. However, when a medium icon is moved into and fixed in the center area 308 of the menu screen 302, the effects processor 244 may change the display parameters for easy viewing. The display parameters can include color, size, brightness, saturation, and/or hue. The display parameters can also include special effects, such as a flashing or blinking action.”);
- **Tu, [0042]** (“Once the selected video is playing, a command/instruction from the controller 114 to bring the menu screen 302 back up invokes the menu screen to be superimposed on top of the currently playing video, as shown in FIG. 4. The attention area 310 of the displayed menu screen 302 shows the video contents icon 340 of the currently playing video shown in the background 330.”);
- **Tu, [0045]** (“The contents display parameter adjustment unit 504 receives the media type from the media type determination unit 502 and operates to generate at least one adjustment value for at least one display parameter of the displayed media contents. The display parameters can include color, size, brightness, saturation, hue, and other related parameters, which may include

the display speed for video. For example, in the illustrated implementation of FIG. 4, the brightness value is generated and used to dim the Singing Quartet video displayed in the background. In some implementations, an adjustment value for other display parameters can be generated to contrast the foreground GUI menu screen from the background media contents display. In other implementations, adjustment values for a plurality of display parameters may be generated.”);

- **Tu, [0049]** (“If the currently displayed media contents item is determined, at 606, to be broadcast or streaming video, then the video is displayed in full motion in the background, at 608. Otherwise, if the currently displayed media contents item is determined, at 610, to be stored video, then the video is displayed in a freeze frame mode in the background, at 612. Otherwise, if the currently displayed media contents item is determined, at 614, to be music animation/graphics, then the animation/graphics is played in slow motion in the background, at 616. Otherwise, if the currently displayed media contents item is determined, at 618, to be a slide presentation, then the last slide transition is completed, at 620. The last slide is then displayed in still image, at 622.”);
- **Tu, [0050]** (“In other implementations, more, fewer, or different display adjustments are available, such as adding text to the screen (e.g., “PAUSE”) or resizing the displayed contents to match a portion of the screen not obscured by the menu. When the menu screen is no longer displayed (e.g., through an exit command), the contents are returned to their former mode of display.”).

228. Tu discloses superimposing a “GUI menu screen” on top of a “dim[med]” background to “contrast the foreground GUI menu screen from the background.” Tu [0045]. Thus, an item of a work process represented on Tu’s menu screen is displayed brighter and more clearly than the background screen. For example, when the GUI switcher is open while music is playing, the music application is “dim[med]” in the background while music is playing and the application switcher, including a GUI preview for the music application, is

presented to allow the user to switch applications, rendering obvious this claim. Tu, Figs. 3–4, [0033], [0045], [0049]–[0050]. As discussed in Section IX.F.1, a person of ordinary skill in the art would have been motivated to further modify Hotelling in view of Hellyar with Tu’s prominently displayed menu icon teachings such that, when one work process is displayed in the background screen, the menu items are displayed and more clearly.

(b) Claim [6]

229. **Claim [6] recites “The method of claim 1, wherein an item of the second work process is displayed relatively brighter or clearer than an item or a screen of a work process other than the second work process.”** This Claim is met by Hotelling in view of Hellyar and Tu.

230. In addition to the teachings from Hotelling and Hellyar that I discussed above in Sections IX.D–IX.E, **Tu discloses an item of the second work process is displayed relatively brighter or clearer than an item or a screen of a work process other than the second work process** (e.g., icon in “center area” is brighter and/or “flashing”). See, e.g., Tu (emphasis added):

• **Tu, Fig. 3**

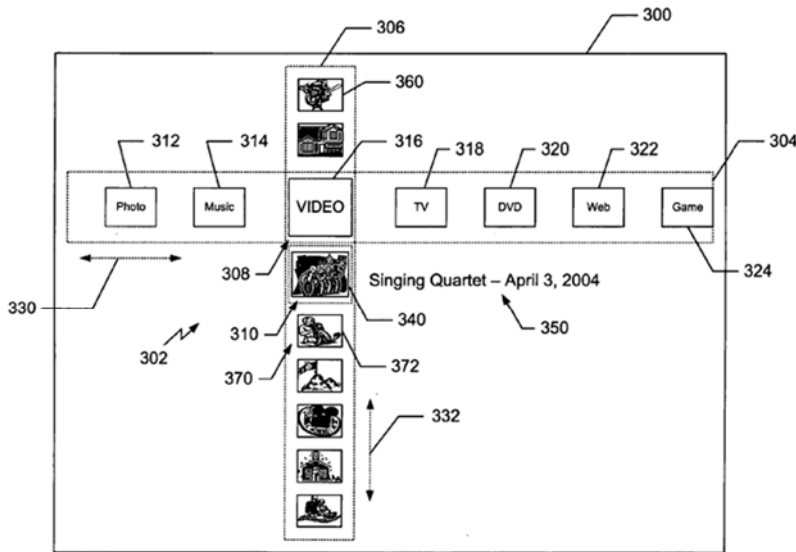


FIG. 3

- **Tu, [0033]** (“As described above, the effects processor 244 displays the medium icons with the same display parameters while the icons are being scrolled. However, when a medium icon is moved into and fixed in the center area 308 of the menu screen 302, the effects processor 244 may change the display parameters for easy viewing. The display parameters can include color, size, brightness, saturation, and/or hue. The display parameters can also include special effects, such as a flashing or blinking action.”);
- **Tu, [0039]** (“In FIG. 3, when the video contents icon array 306 is in a fully extended mode, the array 306 includes eight visible video contents icons representing eight video contents files. In some implementations, the number of visible contents icons can be appropriately varied. Additional video contents icons representing other video contents files can be made visible by scrolling the icons up or down, as shown at 332. Similar to the medium icon processing, the effects processor 244 displays the video contents icons with the same display parameters while the icons are being scrolled. However, when a particular video contents icon 340 is positioned into an attention area 310, below the center area 308, the effects processor 244 changes the display

parameters of the particular video contents icon 340 for easy viewing. In other implementations, the attention area 310 can be positioned at anywhere in the contents icon array 306.”);

- **Tu, [0040]** (“The effects processor 244 enlarges the icon 340 when the icon is positioned into the attention area 310. The display parameters can include color, size, brightness, saturation, and/or hue. The display parameters can also include special effects, such as a flashing or blinking action. Further, when the video contents icon 340 is positioned into the attention area 310, attributes 350 associated with the icon 340 are displayed adjacent to the icon. For example, the attributes 350 can include a title and a recording date.”).

231. In addition to Hellyar’s teachings, as discussed in Section IX.E.2(e), Tu discloses that a selected item in a “center area 308” of the navigation menu shown in Figure 3 are distinguished for “easy viewing” by being shown brighter and/or “flashing.” Tu [0033] (“[W]hen a medium icon is moved into and fixed in the center area 308 of the menu screen 302, the effects processor 244 may change the display parameters . . . includ[ing] color, size, brightness, saturation, and/or hue.”), [0039]–[0040], Fig. 3. For example, when the application preview for Hotelling’s cell-phone (second work process) is the selected window in the center of the screen, that preview will flash and/or be displayed more brightly than the previews for other applications. See Tu [0033]. As discussed in Section IX.F.1, a person of ordinary skill in the art would have been motivated to modify Hellyar’s application window teachings with Tu’s modified display parameter in order to more easily track the selected application preview.

X. SECONDARY CONSIDERATIONS

232. I have not seen any evidence in the prosecution history of the '631 Patent or any related application that I considered of any arguments regarding secondary considerations exist. Nor have I seen anything to indicate that any such evidence would have a nexus to any of the Challenged Claims 1–10. Moreover, as shown above with respect to the prior art referenced in Section IX, any alleged solutions to problems or unexpected results in the '631 patent were already well known.

XI. CONCLUSION

233. In summary, I have concluded that the Challenged Claims are anticipated and rendered obvious over the prior art references as discussed above. Specifically, in my opinion, for at least the reasons above, the Challenged Claims of the '631 Patent are anticipated and rendered obvious as follows.

Ground	Claims	Anticipation or Obviousness	Prior Art References
1	1–10	Anticipated and Obvious	Paasonen
2	1-10	Obvious	Paasonen in view of Nakano
3	4, 7	Obvious	Paasonen in view of Aaltonen
4	4, 7	Obvious	Paasonen in view of Aaltonen and Nakano
5	1–4, 7	Anticipated and Obvious	Hotelling
6	1–4, 6–10	Obvious	Hotelling in view of

Ground	Claims	Anticipation or Obviousness	Prior Art References
			Hellyar
7	5–6	Obvious	Hotelling in view of Hellyar and Tu

234. To the extent it is argued that any further disclosure is required for an element in the Challenged Claims that I have identified as being disclosed, a person of ordinary skill in the art would certainly have found that element obvious to include based on the sample explicit disclosure or inherent disclosures and analysis I have identified above.

235. I reserve the right to supplement my opinions in the future to respond to any arguments that Patent Owner or its expert(s) may raise and to take into account new information as it becomes available to me.

236. In signing this declaration, I recognize that the declaration will be filed as evidence in a contested case before the Patent Trial and Appeal Board of the United States Patent and Trademark Office. I also recognize that I may be subject to cross-examination in the case and that cross-examination will take place within the United States. If cross-examination is required of me, I will appear for cross-examination within the United States during the time allotted for cross-examination.

237. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are

believed to be true; and further 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 Title 18 of the United States Code.

Executed this 10th day of August 2023, in Alpharetta, GA.

Respectfully submitted,



Vijay K. Madiseti, Ph.D.

Appendix A

Dr. Vijay K. Madiseti

Fellow, IEEE

vkm@madiseti.com

Cell: 770-527-0177

Address:

**56 Creekside Park Drive
Johns Creek, GA 30022**

Employment:

- 1989-present: Full Professor of Cybersecurity & Privacy, and Electrical & Computer Engineering (**Georgia Tech, Atlanta, GA 30332**).

Areas of Technical Interest – Computer & Electrical Engineering: Control Systems, Software Engineering, Cloud Computing, Signal Processing, Embedded Systems, Wireless & Mobile Communications, Mobile Applications, Circuit Design (Analog/Digital), Biomedical Instrumentation Hardware/Software, Software Engineering, Digital Signal Processing, Sensors, Blockchain, Cybersecurity, Power Electronics, Wireline & Wireless Computer Networking, Internet of Things (IoT).

Education: **PhD** (EECS, University of California at Berkeley, 1989), **B.Tech** (Hons) in Electronics and Electrical Communications Engineering (Indian Institute of Technology, Kharagpur, 1984).

Startup Companies:

Director, **VP Technologies, Inc.** (1995-): A startup commercialized through Georgia Tech's Advanced Technology Development Corporation (ATDC) focusing on digital software and hardware design services for military market. <http://www.vptinc.net>

Director, **Soft Networks, LLC** (2001-2007): A startup commercialized through Georgia Tech support focusing on software development tools and compilers for Cellular/WiFi/VOIP/telecommunication products. <http://www.soft-networks.com>

Director, **Elastic Video Inc.** (2007- 2009): A startup commercialized through Georgia Tech's VentureLab (<http://venturelab.gatech.edu>) development image and video processing software for wireless & IP networking.

Litigation Experience (2016-2023) With Testimony

(Note: There may be multiple cases between the parties, e.g., District Court v. ITC, US versus Foreign Cases)

Chrimar v. Adtran, Alcatel, et al.

Technology: Power over Ethernet (2015-2017)

Submitted reports & Deposition & Trial

Chamberlain v. Ryobi/TTI

Case No: 1:16-cv-06097 (ND Illinois)

Expert for Ryobi

Technology: Wireless/IoT/Barrier Movement (2016 – 2019)

Submitted reports & deposition & trial testimony

IOEngine v. IMC/Imation

Case No: cv-14-1572-GMS (US Delaware)

Expert for IMC/Imation

Technology: Networked Storage Device (2016-2017)

Submitted reports & deposition & trial testimony

Huawei v. Samsung

Case No: 3:16-cv-2787-WHO (ND Cal)

Expert for Samsung

Technology: 4G/LTE Random Access Protocols (2016-2019)

Submitted reports & deposition

Hitachi Maxell v. ZTE/Huawei

Case No: 5:16-cv-00178-RWS (ED Texas)

Expert for Hitachi Maxell

Technology: Digital Cameras

Submitted reports & deposition (2017 – 2018)

Hitachi Maxell v. Apple

Case No: 5:19-cv-00036-RWS

Expert for Hitachi Maxell

Technology: Digital Cameras

Submitted Expert Reports (2020-)

Qualcomm v. Apple

Case No: 17-ccv-0108-GPC-MDD (SD Cal) Also, Related ITC/FTC Matters

Expert for Qualcomm

Technology: 4G/Wireless Communications/Smartphones (2017-2019)

Submitted Reports and Deposition

Qualcomm v. Apple

Case No: 3:17-cv-01375-DMS-MDD (SD Cal)

Expert for Qualcomm

Technology: 4G/Wireless Communications/Smartphones (2017-2019)

Submitted Reports and Deposition

Optis v. Huawei

Case No: 2:17-cv-123 (E.D. Texas)

Expert for Optis Wireless

Technology: 4G/Video (2017-2019)

Submitted reports & deposition

Beckman Coulter v. Sysmex

Case No: 1:17-cv-24049-DPG (ND Illinois)

Expert for Sysmex

Technology: Medical Instrumentation Automation (2017-present)

Testifying Expert

TQ Delta v. Xyxel/Adtran

Expert for TQ Delta

Technology: DSL Technologies (2018-present)

Testifying Expert

3GL/KPN v. LG, Blackberry, HTC

Expert for 3GL/KPN

Technology: 4G/LTE Protocols (2018-present)

Reports and Deposition / Testifying Expert

Rovi v. Comcast

ITC No. 337-TA-1103 (ITC)

Expert for Rovi

Technology: Digital Video & interactive GUI (Jan 2018-May 2018)

Reports and Deposition

Cirba/Densify v. VMWare

Case No: 1:19-cv-00742-LPS

Expert for Cirba/Densify

Technology: Virtualization (2019-present)

Reports Deposition/PI Hearing / Testifying Expert

Power Integrations v. On Semiconductor

Case No: 17-cv-03189-BLF

Expert for On Semiconductor

Technology: Power Electronics (2018-2019)

Reports & Deposition/Testifying Expert.

Pathway v. Lumens, IPEVO, Aver JDA

Case No: ITC 337-TA-1045

Expert for Lumens JDA

Technology: Document Cameras (2016-2019)

Reports & Deposition (for IPR/ITC, no trial)

Wilan v. Apple

Case No: 3:14-cv-2235

Expert for WiLan

Technology: Voice over LTE/LTE Protocols (2017-2020)

Reports and Testified through Deposition/Trials

St. Lawrence Comm. v. Amazon

Case No. 2:19-CV-00027-JRG

Expert for Amazon

Technology: Multimedia Codecs

Submitted Expert Reports/Deposition (2020-2020)

PanOptis v. Apple

Case No. 2:19-CV-00066-JRG

Expert for PanOptis

Technology: LTE

Submitted Expert Reports/Deposition/Trial (2019-2020)

GAS v. Sprint Communications

Case No. 2:20-cv-00007-RWS

Expert for General Access (GAS)

Technology: LTE

Submitted Expert Reports/Deposition (9/21-)

Causam v. Ecobee, Itron, et al

Case No. ITC 337-TA-1277

Expert for Causum

Technology: Load Management

Submitted Expert Reports and Deposition (6/22)

Masimo v. Apple

Case No. ITC 337-TA-1276 and related cases

Expert for Masimo

Technology: Pulse Oximetry

Submitted Expert Reports, Deposition & Trial

TQ Delta v. Nokia, Adtran, CommScope, 2Wire, et al

Cases in Delaware & Texas

Expert for TQ Delta

Technology: DSL and Cable Modems

Submitted Expert Reports, Depositions & Hearings.

Paltalk Holdings v. Cisco Systems Inc.

Western District of Texas (6:21-CV-00757-ADA)

Expert for Paltalk Holdings

Technology: VOIP Technologies

Submitted Expert Reports and Deposition

Ollnova Technologies v. Carrier Group

Southern District of Florida (Case 9:22-cv-80388-DMM)

Expert for Ollnova

Technology: IoT

Submitted Report and Deposition.

VideoLabs v. Asus, et al

Case No: ITC 337-TA-1323

Expert for VideoLabs

Technology: Video Conferencing

No report or deposition

Additional matters include declarations supporting IPRs at the PTAB for Google (US Patent 8,601,154), On Semiconductor (US Patent 6,212,079), Ubisoft (US Patent 5,490,216), Lumens JDA (2017-2019), Broadcom (WiFi), Sony (US Patent 6,101,534), Kia, (US Patent 5,530,431), Qualcomm, Ericsson, Daimler Benz, VW, Estech, Amazon, Ring, Digital Ally, On Semiconductor, LoganTree, Unified Patents, BMC Software, Masimo, Sentius, BMW, Facebook, PAWS, Google, Daimler, FedEx, and Dolby.

Earned Degrees

1. B. Tech (Hons), Electronics & Electrical Comm. Engineering

Indian Institute of Technology (IIT), Kharagpur, India
1984.

2. Ph.D., Electrical Engineering & Computer Sciences (EECS)

University of California (UC), Berkeley. CA
1989.

Books

1. VLSI Digital Signal Processors

Madiseti, V.K.;

Boston: MA, IEEE Press: Butterworth Heinemann, 1995, 525 pp.

2. Quick-Turnaround ASIC Design in VHDL

Romdhane, M., Madiseti, V.K., Hines, J.

Boston: MA, Kluwer Academic Press, 1996, 190 pp.

3. The Digital Signal Processing Handbook (First Edition)

Madiseti, V. K., Williams, D. (Editors)

CRC Press, Boca Raton, Fla, 1998, 2500 pp.

4. VHDL: Electronics Systems Design Methodologies.

Madiseti, V. K. (Editor)

Boston: MA, IEEE Standards Press, 2000, ISBN 0-7381-1878-8.

5. Platform-Centric Approach to System-on-Chip (SoC) Design.

Madiseti, V. K., Arpnikanondt, A.

Springer, Boston: MA, Springer, 2004, 280 pp.

6. The Digital Signal Processing Handbook – Second Edition.

Madiseti, V. K. (2009)

CRC Press, Boca Raton, Fla.

7. Cloud Computing: A Hands-On Approach

A Bahga, V. Madiseti (2013)

Amazon CreateSpace Publishing, 2013, 454 pp.

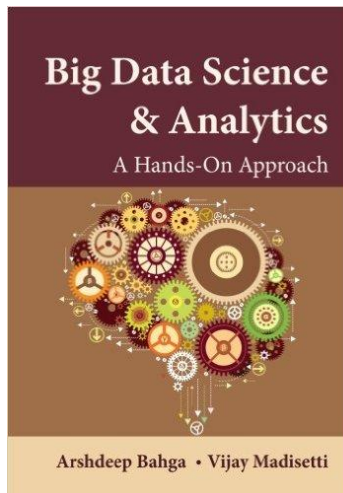
8. Internet of Things: A Hands-On Approach

A Bahga, V. Madiseti (2014)

Amazon CreateSpace Publishing, 2014, 450 pp.



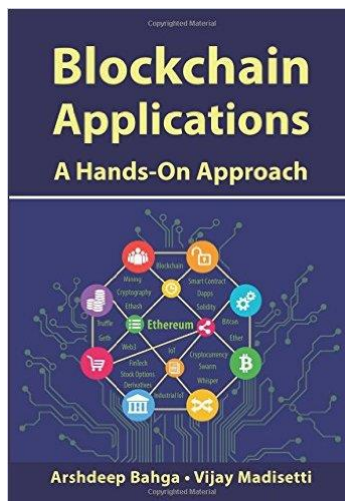
9. Big Data Science & Analytics: A Hands-On Approach
A Bahga, V. Madiseti (2016)
 Amazon CreateSpace Publishing, 2016, 542 pp.



10. Blockchain Applications: A Hands-On Approach

A Bahga, V. Madiseti (2017)

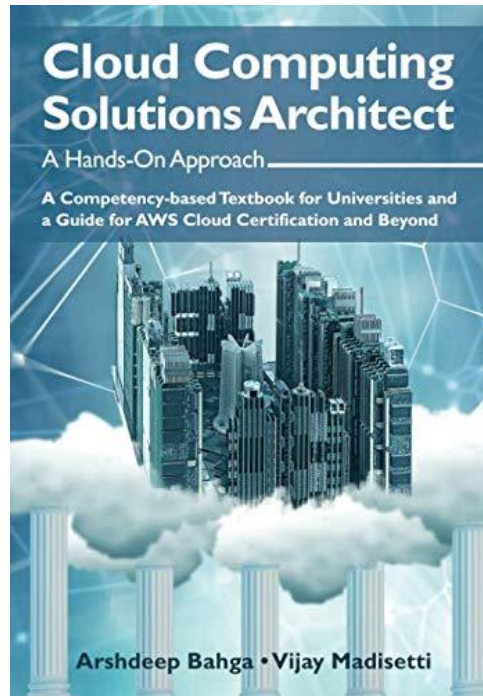
Amazon CreateSpace Publishing, 2017, 380 pp.



11. Cloud Solutions Architect: A Hands-On Approach

A Bahga, V. Madiseti (2020)

Amazon CreateSpace Publishing, 2020, 930 pp.



Edited Books & Collection of Papers

1. **Advances in Parallel & Distributed Simulation**

Madisetti, V.K.; Nicol, D., Fujimoto, R. (Editors)
San Diego, CA: SCS Press, 1991, 200 pp.

2. **Modeling, Analysis, Simulation of Computer & Telecommunications Systems**

Madisetti, V., Gelenbe, E., Walrand, J. W. (Editors)
Los Alamitos: CA, IEEE Computer Society Press, 1994, 425 pp.

3. **Modeling & Simulations on Microcomputers**

Madisetti, V.K. (Editor)
San Diego, CA: SCS Press, 138 pp. 1990.

Editorship of Journals & Transactions

1. IEEE Design & Test of Computers

Special Issue: Reengineering Digital Systems
April – June 1999 (Vol 16, No 2)
Madiseti, V.K (Editor)
Los Alamitos: CA, IEEE Computer Society Press, 1999.

2. IEEE Design & Test of Computers

Special Issue: Rapid Prototyping of Digital Systems
Fall 1996 (Vol 13, No 3)
Madiseti, V., Richards, M. (Editors)
Los Alamitos: CA, IEEE Computer Society Press, 1994, 425 pp.

3. IEEE Transactions on Circuits & Systems II

Associate Editor: 1993-1995.

4. International Journal in Computer Simulation

Associate Editor: 1990-1993

5. International Journal in VLSI Signal Processing

Editorial Board: 1995 - Present

Issued US Patents

- [1] 11,580,540 Methods and systems for smart contracts for security and filtering, Issued February 14, 2023
- [2] 11,556,928 Methods and systems for smart contracts for security and filtering, Issued January 17, 2023
- [3] 11,553,039 Service meshes and smart contracts for zero-trust systems, Issued January 10, 2023
- [4] 11,538,031 Method and system for identity and access management for blockchain interoperability, Issued December 27, 2022
- [5] 11,528,147 Verifying integrity and secure operations of cloud-based software services, Issued December 13, 2022

- [6] 11,526,879 Method and system for zero-knowledge and identity based key management for decentralized applications, Issued December 13, 2022
- [7] 11,494,764 Methods and systems for smart contracts for security and filtering, Issued November 8, 2022
- [8] 11,483,622 Hybrid blockchains and streamchains using non-crypto hashes for securing audio-, video- image-, and speech-based transactions and contracts, Issued October 25, 2022
- [9] 11,379,612 Method and system for securing cloud storage and databases from insider threats and optimizing performance, Issued July 5, 2022
- [10] 11,316,933 Service meshes and smart contracts for zero-trust systems, Issued April 26, 2022
- [11] 11,316,690 Blockchain token-based cloud orchestration architecture for discrete virtual network instances, Issued April 26, 2022
- [12] 11,314,695 Method and system for real-time collaboration and annotation-based action creation and management, Issued April 26, 2022
- [13] 11,283,865 Service meshes and smart contracts for zero-trust systems, Issued March 22, 2022
- [14] 11,250,394 Method and system for blockchain-based gemstone ownership, identity, custody, supply-chain transfer, trading, and secure payments, Issued February 15, 2022
- [15] 11,226,938 Method and system for real-time collaboration and event linking to documents, Issued January 18, 2022
- [16] 11,223,877 Hybrid blockchains and streamchains using non-crypto hashes for securing audio-, video-, image-, and speech-based transactions and contracts, Issued January 11, 2022
- [17] 11,222,331 Method and system for blockchain-based gemstone ownership, identity, custody, supply-chain transfer, trading, and secure payments, Issued January 11, 2022

- [18] 11,212,268 Method and system for identity and access management for blockchain interoperability, Issued December 28, 2021
- [19] 11,080,240 Method and system for real-time collaboration and annotation-based action creation and management, Issued August 3, 2021
- [20] 11,074,650 Method and system for blockchain-based gemstone ownership, identity, custody, supply-chain transfer, trading, and secure payments, Issued July 27, 2021
- [21] 10,983,768 Method and system for persistent helpers for functions as a service (FAAS) in cloud computing environments, Issued April 20, 2021
- [22] 10,853,772 Method and system for exchange of value or tokens between blockchain networks, Issued December 1, 2020
- [23] 10,579,643 Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing, Issued March 3, 2020
- [24] 10,503,927 Method and system for securing cloud storage and databases from insider threats and optimizing performance, Issued December 10, 2019
- [25] 10,460,283 Smart contract optimization for multiparty service or product ordering system, Issued October 29, 2019
- [26] 10,459,946 Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing, Issued October 29, 2019
- [27] 10,402,589 Method and system for securing cloud storage and databases from insider threats and optimizing performance, Issued September 3, 2019
- [28] 10,394,845 Method and system for tuning blockchain scalability for fast and low-cost payment and transaction processing, Issued August 27, 2019

- [29] 10,289,631 Method and system for tuning blockchain scalability for fast and low-cost payment and transaction processing, Issued May 24, 2019
- [30] 10,255,342 Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing, Issued April 9, 2019
- [31] 10,243,743 Tokens or crypto currency using smart contracts and blockchains, Issued March 26, 2019
- [32] 10,204,339 Method and system for blockchain-based combined identity, ownership, integrity and custody management, Issued February 12, 2019
- [33] 10,204,148 Method and system for tuning blockchain scalability, decentralization, and security for fast and low-cost payment and transaction processing, Issued February 12, 2019
- [34] 10,121,143 Method and system for blockchain-based combined identity, ownership, integrity and custody management, Issued November 6, 2018
- [35] 10,102,526 Method and system for blockchain-based combined identity, ownership, integrity and custody management, Issued October 16, 2018
- [36] 10,102,265 Method and system for tuning blockchain scalability for fast and low-cost payment and transaction processing, Issued October 16, 2018
- [37] 9,935,772 Methods and systems for operating secure digital management aware applications, Issued April 3, 2018
- [38] 9,769,213 Method and system for secure digital object management, Issued September 19, 2017

Ranked #8 in the list of issued US patents in the area of blockchain

Ranking in the area of US patents issued in the area of blockchain technology in 2020

Table 3. Patent by assignee

Applicant (Assignee)	Patents	%	Applicant (Assignee)	Patents	%
IBM	47	14.0%	Wells Fargo Bank	4	1.2%
Accenture Global Solutions	28	8.4%	Alibaba	3	0.9%
Bank of America	24	7.2%	Dell	3	0.9%
Capital One Services	12	3.6%	The Bank Of New York Mellon	3	0.9%
Mastercard	11	3.3%	The Toronto-Dominion Bank	3	0.9%
Guardtime IP Holdings Limited	8	2.4%	Verizon Patent And Licensing	3	0.9%
Madiseti Vijay K.	8	2.4%	Winklevoss I.P.	3	0.9%
Black Gold Coin, Inc.	7	2.1%	SAP	2	0.6%
Cisco	7	2.1%	Sony	2	0.6%
Acronis International	5	1.5%	Stampery	2	0.6%
Blockchain Asics	5	1.5%	Uipeco	2	0.6%
Tran Bao	5	1.5%	Vmware	2	0.6%
Intel	4	1.2%	Walmart Apollo	2	0.6%
Northern Trust	4	1.2%	Wiig Rex	2	0.6%
Shocard	4	1.2%	Xage Security	2	0.6%

Refereed Journal Publications

1. Trends in the Electronic Control of Mine Hoists

*Madiseti, V. and Ramlu, M.,
IEEE Transactions on Industry Applications, Vol IA-22, No. 6,
November/December 1986. Pages 1105-1112*

2. Multilevel range/NEXT performance in digital subscriber loops

Brand, G.; Madiseti, V.; Messerschmitt, D.G.;
Communications, Speech and Vision, IEE Proceedings I [see also IEE
Proceedings-Communications] ,Volume: 136 , Issue: 2 , April 1989
Pages:169 – 174

3. Seismic migration algorithms on parallel computers

Madiseti, V.K.; Messerschmitt, D.G.;
Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on], Volume: 39 , Issue: 7 , July 1991
Pages:1642 – 1654

4. Asynchronous algorithms for the parallel simulation of event-driven dynamical systems

Madiseti, V.K.; Walrand, J.C.; Messerschmitt, D.G.:
ACM Transactions on Modeling and Computer Simulation, v 1, n 3, July 1991,
Pages: 244-74

5. Synchronization mechanisms for distributed event-driven computation

Madiseti, V.K.; Hardaker, D.:

ACM Transactions on Modeling and Computer Simulation, v 2, n 1, Jan. 1992,
Pages: 12-51

6. Efficient VLSI Architectures for the Arithmetic Fourier Transform (AFT)

Kelley, B.T.; Madiseti, V.K.;

Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 41 , Issue: 1 , January
1993

Pages:365-378

7. The fast discrete Radon transform. I. Theory

Kelley, B.T.; Madiseti, V.K.;

Image Processing, IEEE Transactions on ,Volume: 2 , Issue: 3 , July 1993

Pages:382 – 400

8. The Georgia Tech digital signal multiprocessor

Barnwell, T.P., III; Madiseti, V.K.; McGrath, S.J.A.;

Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and
Signal Processing, IEEE Transactions on] ,Volume: 41 , Issue: 7 , July 1993

Pages:2471 – 2487

9. The MIMDIX Environment for Parallel Simulation

Madiseti, V.K.; Hardaker, D.; Fujimoto, R.M.:

Journal of Parallel and Distributed Computing, v18, no. 4, August 1993,

Pages: 473-83.

10.LMSGEN: a prototyping environment for programmable adaptive digital filters in VLSI

Romdhane, M.S.B.; Madiseti, V.K.;

Chapter in VLSI Signal Processing, VII, 1994.,

Pages:33 – 42

11. Fixed-point co-design in DSP

Egolf, T.W.; Famorzadeh, S.; Madiseti, V.K.;

Chapter in VLSI Signal Processing, VII, 1994.,

Pages:113 - 126

12. A fast spotlight-mode synthetic aperture radar imaging system

Madiseti, V.K.;

Communications, IEEE Transactions on ,Volume: 42 , Issue: 234 , February-
April 1994

Pages:873 – 876

13. Rapid prototyping on the Georgia Tech digital signal multiprocessor

Curtis, B.A.; Madiseti, V.K.;

Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] ,Volume: 42 , Issue: 3 , March 1994

Pages:649 – 662

14.Low-power signaling in asymmetric noisy channels via spectral shaping

Sipitca, M.; Madiseti, V.K.;

Signal Processing Letters, IEEE, Volume: 1 , Issue: 8 , Aug 1994

Pages:117 – 118

15. A quantitative methodology for rapid prototyping and high-level synthesis of signal processing algorithms

Madiseti, V.K.; Curtis, B.A.;

Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and Signal Processing, IEEE Transactions on] ,Volume: 42 , Issue: 11 , Nov. 1994

Pages:3188 – 3208

16. Computer Simulation of Application-Specific Signal Processing Systems

Casinovi, G.; Madiseti, V.K.;

International Journal in Computer Simulation, Vol. 4, No. 4, Nov 1994.

17. System partitioning of MCMs for low power

Khan, S.A.; Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 12 , Issue: 1 , Spring 1995

Pages:41 – 52

18. Error correcting run-length limited codes for magnetic recording

Jaejin Lee; Madiseti, V.K.;

Magnetics, IEEE Transactions on ,Volume: 31 , Issue: 6 , Nov. 1995

Pages:3084 – 3086

19. Virtual prototyping of embedded microcontroller-based DSP systems

Madiseti, V.K.; Egolf, T.W.;

Micro, IEEE ,Volume: 15 , Issue: 5 , Oct. 1995

Pages:9 – 21

20. Constrained multitrack RLL codes for the storage channel

Lee, J.; Madiseti, V.K.;

Magnetics, IEEE Transactions on ,Volume: 31 , Issue: 3 , May 1995

Pages:2355 – 2364

21. Rapid digital system prototyping: current practice, future challenges

Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996

Pages:12 – 22

22. Conceptual prototyping of scalable embedded DSP systems

Dung, L.-R.; Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996

Pages:54 – 65

23. Advances in rapid prototyping of digital systems

Madiseti, V.K.; Richards, M.A.;

Design & Test of Computers, IEEE ,Volume: 13 , Issue: 3 , Fall 1996

Pages:9

24. Combined modulation and error correction codes for storage channels

Jaejin Lee; Madiseti, V.K.;

Magnetics, IEEE Transactions on ,Volume: 32 , Issue: 2 , March 1996

Pages:509 – 514

25. Model-based architectural design and verification of scalable embedded DSP systems-a RASSP approach

Dung, L.-R.; Madiseti, V.K.; Hines, J.W.;

Chapter in VLSI Signal Processing, IX, 1996.

Pages:147 – 156

26. Low-power digital filter implementations using ternary coefficients

Hezar, R.; Madiseti, V.K.;

Chapter in VLSI Signal Processing, IX, 1996.,

Pages:179 – 188

27. All-digital oversampled front-end sensors

Romdhane, M.S.B.; Madiseti, V.K.;

Signal Processing Letters, IEEE, Volume: 3 , Issue: 2 , Feb. 1996

Pages:38 – 39

28. Modeling COTS components in VHDL

Calhoun, S., Reese, R; Egolf, T., Madiseti, V.K.;

Journal of VLSI Signal Processing, Volume: 14 , Issue: 2 , Nov 1996

Pages: 24 – 31

29. VHDL-Based Rapid Systems Prototyping

Egolf, T.; Madisetti, V.K.;

Journal of VLSI Signal Processing, Volume: 14 , Issue: 2 , Nov 1996

Pages: 40-52

30. Interface design for core-based systems

Madisetti, V.K.; Lan Shen;

Design & Test of Computers, IEEE ,Volume: 14 , Issue: 4 , Oct.-Dec. 1997

Pages:42 - 51

31. Incorporating cost modeling in embedded-system design

Debardelaben, J.A.; Madisetti, V.K.; Gadiant, A.J.;

Design & Test of Computers, IEEE ,Volume: 14 , Issue: 3 , July-Sept. 1997

Pages:24 - 35

32. On homomorphic deconvolution of bandpass signals

Marenco, A.L.; Madisetti, V.K.;

Signal Processing, IEEE Transactions on [see also Acoustics, Speech, and

Signal Processing, IEEE Transactions on] ,Volume: 45 , Issue: 10 , Oct. 1997

Pages:2499 - 2514

33. A case study in the development of multi-media educational material: the VHDL interactive tutorial

Gadiant, A.J.; Stinson, J.A., Jr.; Taylor, T.C.; Aylor, J.H.; Klenke, R.H.;

Salinas, M.H.; Madisetti, V.K.; Egolf, T.; Famorzadeh, S.; Karns, L.N.; Carter, H.W.;

Education, IEEE Transactions on ,Volume: 40 , Issue: 4 , Nov. 1997

Pages:17 pp.

34. Adaptive mobility management in wireless networks

Jeongwook Kim; Madisetti, V.K.;

Electronics Letters ,Volume: 34 , Issue: 15 , 23 July 1998

Pages:1453 - 1455

35. Efficient implementation of two-band PR-QMF filterbanks

Hezar, R.; Madisetti, V.K.;

Signal Processing Letters, IEEE ,Volume: 5 , Issue: 4 , April 1998

Pages:92 - 94

36. On fast algorithms for computing the inverse modified discrete cosine transform

Yun-Hui Fan; Madisetti, V.K.; Mersereau, R.M.;

Signal Processing Letters, IEEE ,Volume: 6 , Issue: 3 , March 1999

Pages:61 - 64

37. System on chip or system on package?

Tummala, R.R.; Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999
Pages:48 – 56

38. Reengineering legacy embedded systems

Madiseti, V.K.; Jung, Y.-K.; Khan, M.H.; Kim, J.; Finnessy, T.;

Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999
Pages:38 – 47

39. Reengineering digital systems

Madiseti, V.K.;

Design & Test of Computers, IEEE ,Volume: 16 , Issue: 2 , April-June 1999
Pages:15 – 16

40. Parameter optimization of robust low-bit-rate video coders

Sangyoun Lee; Madiseti, V.K.;

Circuits and Systems for Video Technology, IEEE Transactions on, Volume: 9
Issue: 6 , Sept. 1999
Pages:849 – 855

41. Closed-form for infinite sum in bandlimited CDMA

Jatunov, L.A.; Madiseti, V.K.;

Communications Letters, IEEE ,Volume: 8 , Issue: 3 , March 2004
Pages:138 – 140

**42. A new protocol to enhance path
reliability and load balancing in mobile ad hoc networks**

Argyriou, A.; Madiseti, V.K.;

Journal of Ad Hoc Networks, Elsevier Press, 2004

43. Closed-form analysis of CDMA systems using Nyquist pulse

Jatunov, L.A.; Madiseti, V. K.;

Communications Letters, IEEE (Under Revision), 2005.

**44. Systematic Design of End-to-End Wireless Mobility Management
Protocols,**

Argyriou, A.; Madiseti, V. K.;

ACM/Springer Wireless Networks (WINET), Accepted 2005.

45. A Novel End-to-End Approach for Video Streaming Over the Internet,

Argyriou, A.; Madiseti, V. K.;

Kluwer Telecommunications Systems, Vol. 28, No. 2, Pages 133-150, Jan
2005. *Special Issue on Multimedia Streaming.*

- 46. An Analytical Framework of RD Optimized Video Streaming with TCP,**
Argyriou, A.; Madiseti, V. K.;
IEEE Transactions on Multimedia, Submitted for review in March 2005.
- 47. Modeling the Effect of Handoffs on Transport Protocol Performance,**
Argyriou, A.; Madiseti, V. K.;
IEEE Transactions on Mobile Computing, Submitted for review in March 2005
- 48. Throughput Models for Transport Protocols with CBR and VBR Traffic Workloads",**
Argyriou, A.; Madiseti, V. K.;
ACM Transactions on Multimedia Computing, Communications & Applications,
Submitted for review in April 2005.
- 49. "Electronic System, Platform & Package Codesign",**
Madiseti, V. K.
IEEE Design & Test of Computers, Volume 23, Issue 3, June 2006. pages
220-233.
- 50. "The Design of an End-to-End Handoff Management Protocol",**
A. Argyriou, Madiseti, V. K.
Wireless Networks, Springer, May 2006.
- 51. "A Soft-Handoff Transport Protocol for Media Flows in Heterogeneous Mobile Networks ",**
A. Argyriou, Madiseti, V. K.
Computer Networks, Vol 50, Issue 11, Pages 1860-1871, August 2006.
- 52. "Computationally Efficient SNR Estimation for Bandlimited WCDMA Systems"**
L. Jatunov, Madiseti, V. K.
IEEE Transactions on Wireless Communications, Volume 5, Issue 13,
December 2006, Pages 3480-3491.
- 53. "Space-Time Codes for Wireless & Mobile Applications",**
M. Sinnokrot, Madiseti, V.K.
DSP Handbook, Second Edition, 2009 (to be published)
- 54. A. Bahga, V. Madiseti, "Rapid Prototyping of Advanced Cloud-Based Systems", IEEE Computer**, vol. 46, no. 11, Nov 2013, pp 76-83, 2013

- 55.A. Bahga, V. Madiseti, "**Cloud-Based Information Integration & Informatics Framework for Healthcare Applications**", *IEEE Computer*, February 2015.
- 56.A. Bahga, V. Madiseti, "**A Cloud-based Approach for Interoperable EHRs**", *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 5, Sep 2013, pp. 894-906, 2013
- 57.A. Bahga, V. Madiseti, "**Cloud-Based Information Technology Framework for Data Driven Intelligent Transportation Systems**", *Journal of Transportation Technologies*, vol.3 no.2, April 2013
- 58.A. Bahga, V. Madiseti, "**Performance Evaluation Approach for Multi-tier Cloud Applications**", *Journal of Software Engineering and Applications*, vol. 6, no. 2, pp. 74-83, Mar 2013.
- 59.Yusuf, A., V. Madiseti, "**Configuration for Predicting Travel Time Using Wavelet Packets and Support Vector Regression**", *Journal of Transportation Technologies*, vol 3, no. 3, June 2013.
- 60.A. Bahga, V. Madiseti, "**Analyzing Massive Machine Maintenance Data in a Computing Cloud**", *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 10, pp. 1831 - 1843, 2012.
- 61.N. Radia. Y. Zhang, M. Tatimapula, V. Madiseti, "**Next Generation Applications on Cellular Networks: Trends, Challenges, and Solutions**," *Proceedings of the IEEE*, Vol 100, Issue 4, pp. 841-854, 2012.
- 62.A. Bahga, V. Madiseti, "**Rapid Prototyping of Advanced Cloud-Based Systems**", *IEEE Computer*, vol. 46, no. 11, Nov 2013, pp 76-83, 2013
- 63.A. Bahga, V. Madiseti, "**A Cloud-based Approach for Interoperable EHRs**", *IEEE Journal of Biomedical and Health Informatics*, vol. 17, no. 5, Sep 2013, pp. 894-906, 2013
- 64.A. Bahga, V. Madiseti, "**Cloud-Based Information Integration & Informatics Framework for Healthcare Applications**", *IEEE Computer*, February 2015.
- 65.A. Bahga, V. Madiseti, "**A Blockchain Platform for Industrial Internet of Things**", *Journal of Software Engineering and Applications*, Vol 9, No. 10, October 2016.

- 66. A. Bahga, V. Madiseti, R. Madiseti, and A. Dugenske, "**Software Defined Things for Manufacturing Networks**", *Journal of Software Engineering and Applications*, Vol 9, No. 9, September 2016.
- 67. A. Bahga, V. Madiseti, J. Corporan, "**Result-as-a-Service (RAAS): Persistent Helper Functions in a Serverless Offering**", *Journal of Software Engineering and Applications*, Vol 13, No 10, October 2020.
- 68. A. Bahga, V. Madiseti, "**A Value-Token Transfer Protocol (VTP) for Decentralized Finance**", *Journal of Software Engineering and Applications*, Vol 13, No. 11, February 2020
- 69. V. Madiseti, S. Panda, "**A Dynamic Leader Election Algorithm for Decentralized Networks**", *Journal of Transportation Technologies*, Vol 11, No 3, July 2021

Peer Reviewed Conference Publications

- 1. **Dynamically-reduced complexity implementation of echo cancelers**
Madiseti, V.; Messerschmitt, D.; Nordstrom, N.;
Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86, Volume: 11, Apr 1986
- 2. **Seismic migration algorithms using the FFT approach on the NCUBE multiprocessor**
Madiseti, V.K.; Messerschmitt, D.G.;
Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on, 11-14 April 1988
- 3. **Seismic migration algorithms on multiprocessors**
Madiseti, V.K.; Messerschmitt, D.G.;
Acoustics, Speech, and Signal Processing, 1988. ICASSP-88., 1988 International Conference on, 11-14 April 1988
Pages: 2124 - 2127 vol.4
- 4. **WOLF: A rollback algorithm for optimistic distributed simulation systems**
Madiseti, V.; Walrand, J.; Messerschmitt, D.;
Simulation Conference Proceedings, 1988 Winter, December 12-14, 1988
Pages: 296 - 305
- 5. **Efficient distributed simulation**
Madiseti, V.; Walrand, J.; Messerschmitt, D.;
Simulation Symposium, 1989. The 22nd Annual, March 28-31, 1989
Pages: 5 - 6

- 6. High speed migration of multidimensional seismic data**
Kelley, B.; Madiseti, V.;
Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991
International Conference on, 14-17 April 1991
Pages:1117 - 1120 vol.2
- 7. Performance of a fast analog VLSI implementation of the DFT**
Buchanan, B.; Madiseti, V.; Brooke, M.;
Circuits and Systems, 1992., Proceedings of the 35th Midwest Symposium
on, 9-12 Aug. 1992
Pages:1353 - 1356 vol.2
- 8. Task scheduling in the Georgia Tech digital signal multiprocessor**
Curtis, B.A.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE
International Conference on, Volume: 5, 23-26 March 1992
Pages:589 - 592 vol.5
- 9. The fast discrete Radon transform**
Kelley, B.T.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE
International Conference on, Volume: 3, 23-26 March 1992
Pages:409 - 412 vol.3
- 10. Yield-based system partitioning strategies for MCM and ASEM design**
Khan, S.; Madiseti, V.;
Multi-Chip Module Conference, 1994. MCMC-94, Proceedings., 1994 IEEE, 15-
17 March 1994
Pages:144 - 149
- 11. Multitrack RLL codes for the storage channel with immunity to intertrack interference**
Lee, J.; Madiseti, V.K.;
Global Telecommunications Conference, 1994. GLOBECOM '94.
'Communications: The Global Bridge', IEEE, Volume: 3, 28 Nov.-2 Dec. 1994
Pages:1477 - 1481 vol.3
- 12. A parallel mapping of backpropagation algorithm for mesh signal processor**
Khan, S.A.; Madiseti, V.K.;
Neural Networks for Signal Processing [1995] V. Proceedings of the 1995
IEEE Workshop, 31 Aug.-2 Sept. 1995
Pages:561 - 570

13. Virtual prototyping of embedded DSP systems

Madiseti, V.K.; Egolf, T.; Famorzadeh, S.; Dung, L.-R.;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on, Volume: 4 , 9-12 May 1995
Pages:2711 - 2714 vol.4

14. Assessing and improving current practice in the design of application-specific signal processors

Shaw, G.A.; Anderson, J.C.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on ,Volume: 4 , 9-12 May 1995
Pages:2707 - 2710 vol.4

15. Introduction to ARPA's RASSP initiative and education/facilitation program

Corley, J.H.; Madiseti, V.K.; Richards, M.A.;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on, Volume: 4 , 9-12 May 1995
Pages:2695 - 2698 vol.4

16. DSP design education at Georgia Tech

Madiseti, V.K.; McClellan, J.H.; Barnwell, T.P., III;
Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995
International Conference on, Volume: 5 , 9-12 May 1995
Pages:2869 - 2872 vol.5

17. Rapid prototyping of DSP systems via system interface module generation

Famorzadeh, S.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference
Proceedings., 1996 IEEE International Conference on, Volume: 2 , 7-10 May
1996
Pages:1256 - 1259 vol. 2

18. Rapid prototyping of DSP chip-sets via functional reuse

Romdhane, M.S.B.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference
Proceedings., 1996 IEEE International Conference on ,Volume: 2 , 7-10 May
1996
Pages:1236 - 1239 vol. 2

19. A constructive deconvolution procedure of bandpass signals by homomorphic analysis

Marenco, A.L.; Madiseti, V.K.;
Geoscience and Remote Sensing Symposium, 1996. IGARSS '96. 'Remote

Sensing for a Sustainable Future.', International, Volume: 3, 27-31 May 1996
Pages:1592 - 1596 vol.3

20. BEEHIVE: an adaptive, distributed, embedded signal processing environment

Famorzadeh, S.; Madiseti, V.; Egolf, T.; Nguyen, T.;
Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE
International Conference on, Volume: 1, 21-24 April 1997
Pages:663 - 666 vol.1

21. Target detection from coregistered visual-thermal-range images

Perez-Jacome, J.E.; Madiseti, V.K.;
Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE
International Conference on, Volume: 4, 21-24 April 1997
Pages:2741 - 2744 vol.4

22. Variable block size adaptive lapped transform-based image coding

Klausutis, T.J.; Madiseti, V.K.;
Image Processing, 1997. Proceedings., International Conference on ,Volume:
3 , 26-29 Oct. 1997
Pages:686 - 689 vol.3

23. A Rate 8/10 (0, 6) MTR Code And Its Encoder/decoder

Jaejin Lee; Madiseti, V.K.;
Magnetics Conference, 1997. Digests of INTERMAG '97., 1997 IEEE
International, 1-4 April 1997
Pages:BS-15 - BS-15

24. VHDL models supporting a system-level design process: a RASSP approach

DeBardelaben, J.A.; Madiseti, V.K.; Gadiant, A.J.;
VHDL International Users' Forum, 1997. Proceedings, 19-22 Oct. 1997
Pages:183 - 188

25. A performance modeling framework applied to real time infrared search and track processing

Pauer, E.K.; Pettigrew, M.N.; Myers, C.S.; Madiseti, V.K.;
VHDL International Users' Forum, 1997. Proceedings, 19-22 Oct. 1997
Pages:33 - 42

26. System design and re-engineering through virtual prototyping: a temporal model-based approach

Khan, M.H.; Madiseti, V.K.;
Signals, Systems & Computers, 1998. Conference Record of the Thirty-
Second Asilomar Conference on, Volume: 2 , 1-4 Nov. 1998
Pages:1720 - 1724 vol.2

- 27. Web-Enabled Information Appliances for Broadband Residential Networks”, F. Ahmed, V. Madisetti, Y. Jiao, V. Dasigi, Yamacraw IAB Presentation, 2000. Atlanta, GA**
- 28. Web-Enabled Transcoding for Broadband Residential Networks (BRAN), M. Rahim, V. Madisetti, K. Naik, V. Dasigi, Yamacraw IAB Presentation, 2000, Atlanta, GA.**
- 29. Enforcing Interdependencies and Executing Transactions Atomically Over Autonomous Mobile Data Stores Using SyD Link Technology, S. K. Prasad, V. Madisetti, et al, IEEE International Conference on Distributed Computing, 2003.**
- 30. A debugger RTOS for embedded systems**
Akgul, T.; Kuacharoen, P.; Mooney, V.J.; Madisetti, V.K.;
Euromicro Conference, 2001. Proceedings. 27th, 4-6 Sept. 2001
Pages:264 - 269
- 31. Adaptability, extensibility and flexibility in real-time operating systems**
Kuacharoen, P.; Akgul, T.; Mooney, V.J.; Madisetti, V.K.;
Digital Systems, Design, 2001. Proceedings. Euromicro Symposium on, 4-6 Sept. 2001
Pages:400 – 405
- 32. Effect of handoff delay on the system performance of TDMA cellular systems**
Turkboylari, M.; Madisetti, V.K.;
Mobile and Wireless Communications Network, 2002. 4th International Workshop on, 9-11 Sept. 2002
Pages:411 – 415
- 33. Enforcing interdependencies and executing transactions atomically over autonomous mobile data stores using SyD link technology**
Prasad, S.K.; Bourgeois, A.G.; Dogdu, E.; Sunderraman, R.; Yi Pan; Navathe, S.; Madisetti, V.;
Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on, 19-22 May 2003
Pages:803 – 809
- 34. Performance evaluation and optimization of SCTP in wireless ad-hoc networks**
Argyriou, A.; Madisetti, V.;
Local Computer Networks, 2003. LCN '03. Proceedings. 28th Annual IEEE International Conference on, 20-24 Oct. 2003
Pages:317 - 318

35. Implementation of a calendar application based on SyD coordination links

Prasad, S.K.; Bourgeois, A.G.; Dogdu, E.; Sunderraman, R.; Yi Pan; Navathe, S.; Madiseti, V.;
Parallel and Distributed Processing Symposium, 2003. Proceedings.
International, 22-26 April 2003
Pages:8 pp.

36. Bandwidth aggregation with SCTP

Argyriou, A.; Madiseti, V.;
Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE Volume:
7, 1-5 Dec. 2003
Pages:3716 - 3721 vol.7

37. Software streaming via block streaming

Kuacharoen, P.; Mooney, V.J.; Madiseti, V.K.;
Design, Automation and Test in Europe Conference and Exhibition, 2003
Pages:912 – 917

38. Frequency-dependent space-interleaving for MIMO OFDM systems

Mohajerani, P.; Madiseti, V.K.;
Radio and Wireless Conference, 2003. RAWCON '03. Proceedings, Aug. 10-13, 2003
Pages:79 - 82

39. A media streaming protocol for heterogeneous wireless networks

Argyriou, A.; Madiseti, V.;
Computer Communications, 2003. CCW 2003. Proceedings. 2003 IEEE 18th Annual Workshop on, 20-21 Oct. 2003
Pages:30 – 33

40. Realizing load-balancing in ad-hoc networks with a transport layer protocol

Argyriou, A.; Madiseti, V.;
Wireless Communications and Networking Conference, 2004. WCNC. 2004 IEEE, Volume: 3, 21-25 March 2004
Pages:1897 - 1902 Vol.3

41. Streaming H.264/AVC video over the Internet

Argyriou, A.; Madiseti, V.;
Consumer Communications and Networking Conference, 2004. CCNC 2004. First IEEE, 5-8 Jan. 2004
Pages:169 – 174

Other Publications

1. **A Transport Layer Technology for Improving the QoS of Networked Multimedia Applications <draft-madiseti-arguriou-qos-sctp-00.txt>.**
Madiseti, V., Arguriou, A.
IETF Internet-Draft, Jul 25, 2002.
2. **Voice & Video over Mobile IP Networks <draft-madiseti-arguriou-voice-video-mip-00.txt>**
Madiseti, V., Arguriou, A.
IETF Internet-Draft, Nov 20, 2002.
3. **Enhancements to ECRTTP with Applications to Robust Header Compression for Wireless Applications. <draft-madiseti-rao-suresh-rohc-00.txt>**
Madiseti, V.; Rao, S., Suresh, N.
IETF Internet-Draft, June 30, 2003.

Ph.D. Students Graduated

1. **Brian T. Kelley, 1992**
VLSI Computing Architectures for High Speed Signal Processing
Member of Technical Staff, Motorola

Winner of Dr. Thurgood Marshall Dissertation Fellowship Award
2. **Bryce A. Curtis, 1992**
Special Instruction Set Multiple Chip Computer for DSP
Member of Technical Staff, IBM
3. **Jaejin Lee, 1994**
Robust Multitrack Codes for the Magnetic Channel
Professor, Yonsei University, Korea
4. **Mohamed S. Ben Romdhane, 1995**
Design Synthesis of Application-Specific IC for DSP
Director of IP, Rockwell
5. **Shoab A. Khan, 1995**
Logic and Algorithm Partitioning on MCMs
Professor, National University of Science & Technology, Pakistan
6. **Lan-Rong Dung, 1997**
VHDL-based Conceptual Prototyping of Embedded DSP Architectures
Professor, National Chaio Tung University, Taiwan

Winner of VHDL International Best PhD Thesis Award, 1997

- 7. Thomas W. Egolf, 1997**
Virtual Prototyping of Embedded DSP Systems
Distinguished Member of Technical Staff, Agere
- 8. Alvaro Marenco, 1997**
On Homomorphic Deconvolution of Bandpass Signals
Professor, Texas A&M University

Winner of GIT ECE Outstanding Teaching Assistant Award
- 9. Shahram Famorzadeh, 1997**
BEEHIVE: A Distributed Environment for Adaptive Signal Processing
Member of Technical Staff, Rockwell
- 10. Timothy J. Klausutis, 1997**
Adaptive Lapped Transforms with Applications to Image Coding.
US Air Force/Univ. of Florida
- 11. Lan Shen, 1998**
Temporal Design of Core-Based Systems
Member of Technical Staff, IBM
- 12. James DeBardelaben, 1998**
Optimization Based Approach to Cost Effective DSP Design
Research Scientist, Johns Hopkins University

Georgia Tech ECE Faculty Award
- 13. Sangyoun Lee, 1999**
Design of Robust Video Signal Processors
Professor, Yonsei University

US Army Sensors Lab Research Excellence Award, 1999
- 14. Rahmi Hezar, 2000**
Oversampled Digital Filters
Member of Technical Staff, Texas Instruments
- 15. Yong-kyu Jung, 2001**
Model-Based Processor Synthesis
Professor, Texas A&M University
- 16. Mustafa Turkboylari, 2002**
Handoff Algorithms for Wireless Applications
Member of Technical Staff, Texas Instruments
- 17. Yun-Hui Fan, 2002**
A Stereo Audio Coder with Nearly Constant Signal to Noise Ratio
Post-Doctoral Research Associate, Northeastern University
- 18. Subrato K. De, 2002**
Design of a Retargetable Compiler for DSP

Member of Technical Staff, Qualcomm

US Army Sensors Lab Research Excellence Award, 1999

19. Chonlameth Aripnikanondt, 2004

System-on-Chip Design with UML

Professor, King Mongkut's University, Thailand

US Army Sensors Lab Research Excellence Award, 1999

20. Loran Jatunov, 2004

Performance Analysis of 3G CDMA Systems

Senior Research Scientist, Soft Networks, LLC. Now at Samsung

21. Antonios Argyiou, 2005, Research Scientist, Philips

22. Pilho Kim, 2009, Scientist, VP Technologies, Inc.

23. M. Sinnokrot, 2009, Staff Engineer, Apple

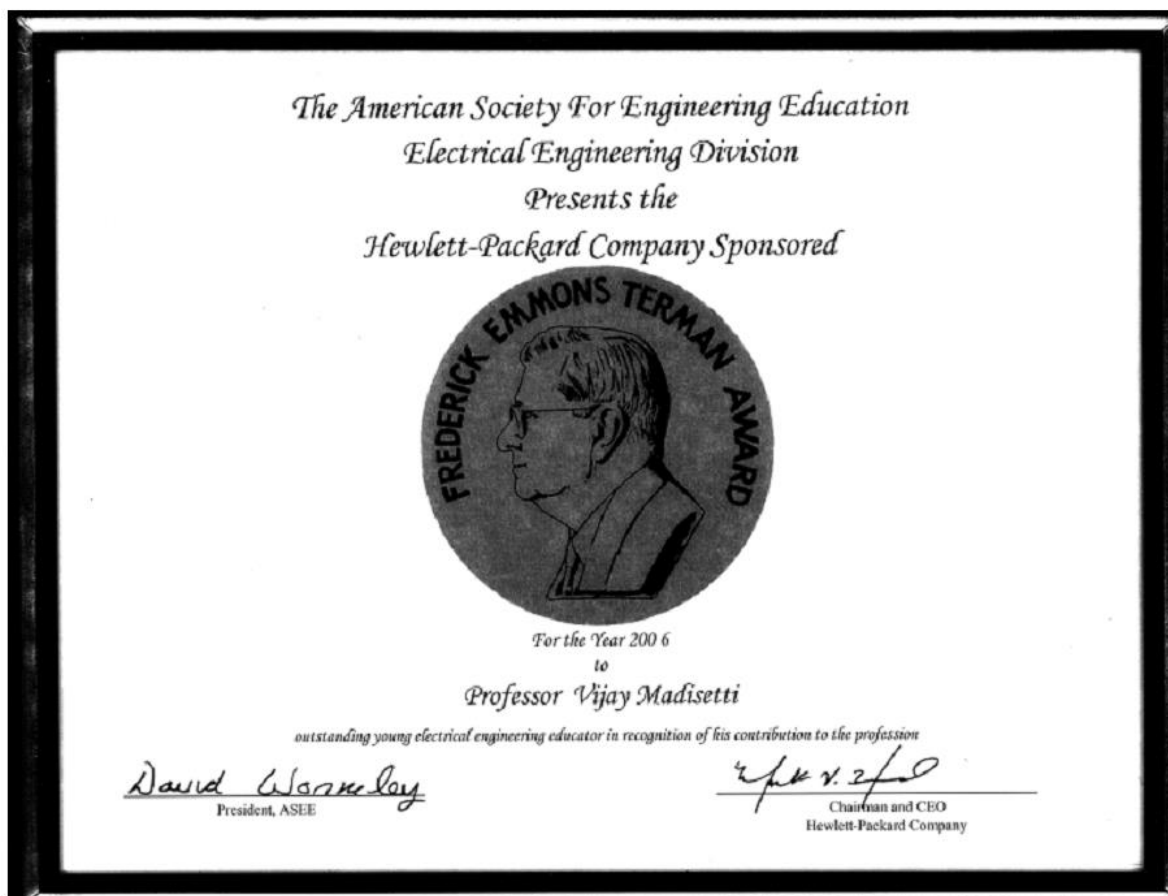
24. A. Yusuf, 2013, Professor, National University, Pakistan

Awards & Honors

1. **Jagasdis Bose National Science Talent Fellowship**, Indian Institute of Technology, Kharagpur, 1980-1984
2. **General Proficiency Prize**, Indian Institute of Technology, Kharagpur, 1984
3. **Demetri Angelakos Outstanding Graduate Student Award**, Univ. of California, Berkeley, 1989
4. **Ira Kay IEEE/ACM Best Paper Award** for Best Paper presented at IEEE Annual Simulation Symposium, 1989
5. **IBM Faculty Development Award** 1990
6. **Technical Program Chair**, IEEE Workshop on Parallel and Distributed Simulation. 1990
7. **Technical Program Chair**, IEEE MASCOTS'94
8. **NSF RI Award**, 1990
9. **VHDL International Best PhD Dissertation Advisor Award**, 1997
10. **Georgia Tech Outstanding Doctoral Dissertation Advisor Award**, 2001

11. ASEE 2006 Frederick Emmons Terman Medal, 2006

12. Fellow of IEEE



Intellectual Property Disclosures (Georgia Tech)

<u>Patent</u>	<u>Date</u>	<u>Description</u>
2843	2004	Method and Apparatus for Improving the Performance of Wireless LANs
2825	2003	Method and Apparatus for Optimal Partitioning and Ordering of Antennas for Layered Space-Time Block Codes in MIMO Communications Systems
2815	2003	How to Rapidly Develop a SyD Application
GSU-023	2003	Rapid Development of SyD Applications
2810	2003	System on Mobile Devices Middleware Design
2718	2003	A Transport Layer Algorithm for Improved Anycast Communication

2717	2003	A Novel Transport Layer Load-Balancing Algorithm
2716	2003	A Transport Layer QoS Algorithm
2715	2003	A Novel Transport Layer Algorithm for MPLS Performance
2659	2002	A New Algorithm and Technology for Implementing Mobile IP with Applications to Voice and Video over Mobile IP
2656	2002	Debugging with Instruction-Level Reverse Execution
2655	2002	Embedded Software Streaming
2539		System of Databases: An Enabling Technology for Programming
2517	2002	A Dynamic Instantiated Real-Time Operating System Debugger
2516	2002	A Dynamic Real-Time Operating System
GSU-009	2001	System of Databases: Architecture,, Global Queries, Triggers and Constraints
2480	2001	Mobile Fleet Application based on SyD Technology
2479	2001	System of Databases: A model with coordination links and a calendar application
1893	1999	Beehive
1726	1995	Very High Scale Integrated Circuit Hardware Description Language Models (VHDL Models)
1401	1995	Self-Compensation Receiver (SCR)

Appendix B

List of Materials Considered

Exhibit ("Ex.")	Description
1001	U.S. Patent No. 9,575,631 ("631 Patent")
1002	File History of U.S. Patent No. 9,575,631 ("631 File History")
1005	U.S. Patent No. 9,785,329 ("Paasonen")
1006	U.S. Patent Publication No. 2004/0100479 ("Nakano")
1007	U.S. Patent Publication No. 2006/0020904 ("Aaltonen")
1008	U.S. Patent Publication No. 2005/0257169 ("Tu")
1009	U.S. Patent Publication No. 2006/0197753 ("Hotelling")
1010	U.S. Patent Provisional App. No. 60/658,777 ("Hotelling-Prov.")
1011	U.S. Patent Publication No. 2006/0026521 ("Hotelling-521")
1012	U.S. Patent Publication No. 2003/0117440 ("Hellyar")
1013	U.S. Patent Publication No. 2004/0166895 ("Koenck")
1014	U.S. Patent Publication No. 2004/0219901 ("Loke")
1015	U.S. Patent Publication No. 2002/0068558 ("Janik")
1016	U.S. Patent Publication No. 2005/0110768 ("Marriott")
1017	P.C.T. Patent App. Publication No. W.O. 2000/0065429 ("Törnqvist")
1018	U.S. Patent Publication No. 2003/0076306 ("Zadesky")
1019	U.S. Patent Publication No. 2003/0083106 ("Seo")
1020	European Patent App. Publication No. E.P. 1531607 ("Hon")
1021	U.S. Patent Publication No. 2004/0259542 ("Viitamaki")
1022	P.C.T. Patent App. Publication No. W.O. 2005/064441 and certified

Exhibit ("Ex.")	Description
	translation thereof ("Lee")
1023	U.S. Patent Publication No. 2002/0109678 ("Marmolin")
1024	U.S. Patent No. 5,715,417 ("Gardien")
1025	U.S. Patent Publication No. 2006/0101347 ("Runov")
1026	U.S. Patent Publication No. 2006/0211454 ("Park")
1027	Joint Claim Construction and Prehearing Statement, <i>Pantech Corp. et al. v. LG Electronics Inc. et al.</i> , Case 5:22-cv-00113-RWS-JBB (E.D. Texas), Dkt. No. 118.
1030	File History of U.S. Patent App. No. 11/495,863 ("863-App. File History")
1031	Plaintiff's Patent Rule 3-1 and 3-2 Infringement Contentions, Appendix 9: Preliminary Infringement Contentions of U.S. Patent No. 9,575,631, <i>Pantech Corp. et al. v. LG Electronics Inc. et al.</i> , Case 5:22-cv-00113-RWS-JBB (E.D. Texas) (February 15, 2023).
1032	Claim Construction Expert Report of Charles Mauro CHFP Concerning Certain Terms of U.S. Patent Nos. 7,283,839 And 9,575,631, <i>Pantech Corp. et al. v. LG Electronics Inc. et al.</i> , Case 5:22-cv-00113-RWS-JBB (E.D. Texas) (June 23, 2023).
1033	Redline Comparing the Text of Hotelling-Prov. and Hotelling

Appendix C

LIST OF CHALLENGED CLAIMS:

[1.pre] A method for controlling a predetermined function of a second work process during a first work process in a multitasking mobile terminal that performs the first work process and the second work process, the method comprising:

[1.a] displaying one icon in a designated portion of a display, wherein the displayed icon corresponds to the first work process;

[1.b] if the displayed icon corresponding to the first work process is selected, displaying a screen of the first work process in the display;

[1.c] in response to a first command for sequentially displaying icons, performing sequential display of icons such that an icon corresponding to the second work process is displayed, wherein the icon corresponding to the second work process is displayed in the designated portion of the display;

[1.d] in response to a second command for changing to the second work process, replacing the screen of the first work process with a screen of the second work process;

[1.e] controlling the multitasking of the second work process and the first work process;

[1.f] in response to a third command for activating a control menu for controlling a predetermined function of the first work process, activating the control menu for controlling the predetermined function of the first work process; and

[1.g] in response to a fourth command for controlling the predetermined function of the first work process through the activated control menu, controlling the predetermined function of the first work process through the activated control menu.

[2.pre] The method of claim 1, wherein the controlling the predetermined function of the first work process comprises:

[2.a] generating a control signal for controlling the predetermined function of the first work process in response to the fourth command; and

[2.b] controlling the predetermined function of the first work process in response to the generated control signal.

[3] The method of claim 1, further comprising: in response to a fifth command for deactivating the activated control menu, deactivating the activated control menu.

[4] The method of claim 1, wherein the first work process is an MP3 play process.

[5] The method of claim 1, wherein an item of the first work process is displayed relatively brighter or clearer than the screen of the first work process.

[6] The method of claim 1, wherein an item of the second work process is displayed relatively brighter or clearer than an item or a screen of a work process other than the second work process.

[7] The method of claim 3, wherein the third command and the fifth command are inputted through at least one of a predetermined key, a user's voice, and a motion of the mobile terminal.

[8] The method of claim 1, wherein, in response to a repetition of the first command, performing sequential display of icons such that a displayed icon disappears from a screen of the mobile terminal and an icon appears on the screen of the mobile terminal.

[9] The method of claim 1, wherein an icon to be displayed in the designated portion is replaced with an icon that is different from the icon corresponding to the second work process in response to the first command.

[10] The method of claim 1, wherein the displayed icon is displayed together with at least a part of a screen image of a work process in the display.